

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
„КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМ. ІГОРЯ
СІКОРСЬКОГО»

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

Дипломна робота

на тему: Розробка системи активного автотиматизованого збору даних користувачів соціальної мережі

Студент групи ТІ-61 Бакуна Богдана Валерійовича _____
(прізвище, ім'я, по батькові) (підпис)

Керівник роботи доцент Коваль О.В. _____
(вчені ступінь та звання, прізвище, ініціали) (підпис)

Кількість сторінок 63

Кількість ілюстрацій 26

Київ – 2020

**Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”**

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти перший, бакалаврський

Напрямок підготовки 121 “Інженерія програмного забезпечення”

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ О.В. Коваль

(підпис)

” ____ ” _____ 2020р.

ЗАВДАННЯ

на дипломну роботу студенту

_____ Бакун Богдан Валерійович _____

(прізвище, ім'я, по батькові)

1. Тема роботи _____ Розробка системи активного автоматизованого збору даних користувачів соціальної мережі _____

керівник роботи _____ Коваль Олександр Васильович к.т.н., доцент _____

(прізвище, ім'я, по батькові науковий ступінь, вчене звання)

затверджена наказом вищого навчального закладу від ” ____ ” ____ 201__р. № ____

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи див. Керівництво користувача _____

4.Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити)

1.ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

2.ЗАСОБИ РОЗРОБКИ

3.ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

5. Перелік ілюстративного матеріалу

19 зображень

7 знімків екрану виконання додатку

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання ”__”_____201__р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання дипломної роботи	Термін виконання	Примітки
---	--	------------------	----------

з/п		етапів роботи	
1.	Вивчення та аналіз задачі		
2.	Розробка архітектури та загальної структури системи		
3.	Розробка структур окремих підсистем		
4.	Програмна реалізація системи		
5.	Оформлення пояснювальної записки		
6.	Захист програмного продукту	11.06.2020	
7.	Передзахист	11.06.2020	
8.	Захист	15.06.2020	

Студент _____ Бакун Б.В.
 (підпис) (прізвище та ініціали,)

Керівник роботи _____ Коваль О.В.
 (підпис) (прізвище та ініціали,)

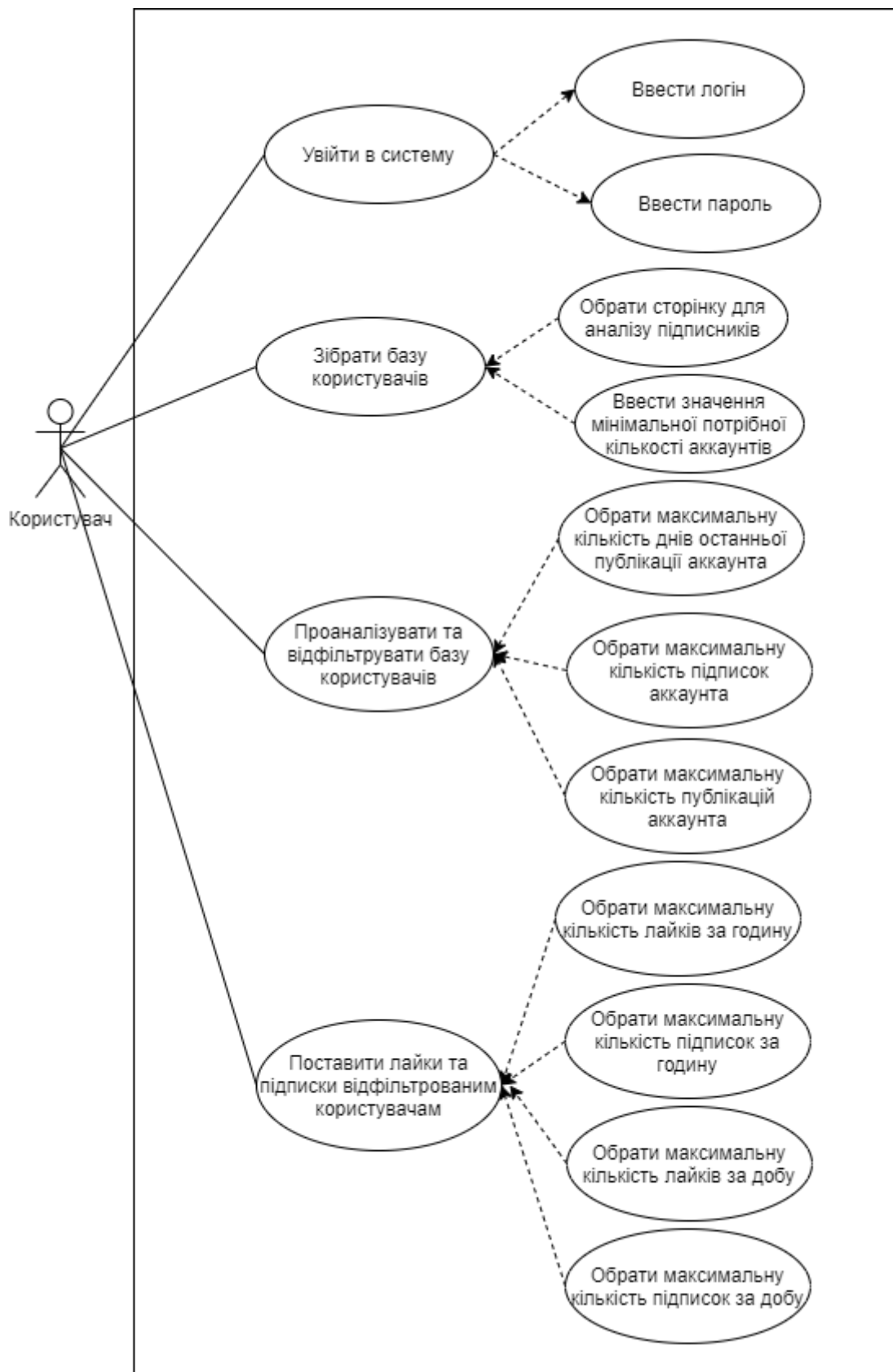


Рисунок 1.1 – Use Case діаграма програмного додатку

АНОТАЦІЯ

Актуальність теми: В наш час користування соціальними мережами бурхливо зростає кожного дня. Все частіше люди переходять до праці в інтернеті, використовуючи соціальні мережі. Користувачи все більше потребують програмного забезпечення, яке займається активним автоматизованим збором даних про інших користувачів – потенційних клієнтів.

Активний режим збору даних визначається як режим збору, при якому ми самостійно ініціюємо запит на інформацію, в якій ми зацікавлені.

Програмний продукт реалізований за допомогою мови програмування Python у середовищі розробки Microsoft Visual Studio Code.

Отримані результати:

- досліджено сучасний метод активного автоматизованого збору даних за допомогою пошуку елементів на сторінці по XPath.

- розроблено програмне забезпечення(бот), яке активно автоматизовано збирає дані про аккаунти в Instagram, відфільтровує небажані аккаунти(Користувач задає потрібні параметри) та забезпечує приток нових підписників, котрі можуть бути зацікавлені у тому, що пропонує користувач бота.

ABSTRACT

Relevance of the topic: Nowadays, the use of social networks is growing rapidly every day. Increasingly, people are turning to the Internet using social media. Users increasingly need software that actively collects automated data about other potential customers.

Active data collection mode is defined as a collection mode in which we independently initiate a request for information in which we are interested.

The software product is implemented using the Python programming language in the Microsoft Visual Studio Code development environment.

The results obtained:

- the modern method of active automated data collection by means of search of elements on page on XPath is investigated.

- developed software (bot), which actively automatically collects data about Instagram accounts, filters out unwanted accounts (User sets the necessary parameters) and provides an influx of new subscribers who may be interested in what the bot user offers.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	8
ВСТУП.....	10
1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ.....	12
1.1 Способи збору даних	13
1.2 Контент-аналіз.....	14
1.2.1 Види контент-аналізу	14
1.2.2 Переваги та недоліки.....	15
1.3 Соціальна мережа Instagram	16
1.3.1 Аналіз Instagram.....	17
1.3.2 Що зробило Instagram таким популярним?	20
1.3.3 Що таке аккаунт блог в Instagram?.....	22
1.3.4 Використання штучного інтелекту в Instagram.....	26
1.4 Обмеження Instagram, які вплинули на реалізацію ПП.....	28
1.4.1 Дослідження лімітів Instagram.....	28
1.4.2 Які санкції можуть бути застосовані за порушення?	31
1.5 Навіщо збирати дані Instagram-користувачів?.....	33
1.5.1 Активний автоматизований збір даних	34
2. ЗАСОБИ РОЗРОБКИ.....	35
2.1 Python.....	36
2.2 Selenium.....	39
2.2.1 Мова програмування для Selenium	39
3. ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ.....	42

3.1 Робочий процес ПЗ	42
3.1.1 Фільтрування бази ЦА.....	44
3.2 Масфоловінг та маслайкінг.....	48
3.3 Використані методи.....	50
3.3.1 Пошук по XPath	50
3.3.2 Використання XML	53
3.3.3 Інші методи	55
3.4 Керівництво користувача	58
3.5 Можливі вдосконалення.....	60
ВИСНОВКИ.....	61
ПЕРЕЛІК ПОСИЛАНЬ.....	62
ДОДАТОК 1.....	64
ДОДАТОК 2.....	67
ДОДАТОК 3.....	79

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

AI (англ. Artificial intelligence) – штучний інтелект.

SEO(англ. Artificial intelligence) – фахівець, що використовує оптимізацію пошуку.

ШІ – штучний інтелект.

СМ – соціальна мережа.

ПП – програмний продукт.

ПЗ – програмне забезпечення.

ЦА – цільова аудиторія.

ВСТУП

Що таке штучний інтелект?

Штучний інтелект - це широка галузь інформатики, завданням якої є побудова розумних машин, спроможних виконувати завдання, що зазвичай потребують людського інтелекту. Штучний інтелект - це міждисциплінарна наука з багатьма підходами, але прогрес у машинному та глибокому навчанні створює зміну парадигми практично у кожному секторі технологічної галузі.

Метою штучного інтелекту є позитивний вплив на діяльність компаній, людей, країн (тощо). Це сприятливо мотивує на дослідження в різних сегментах та сферах діяльності – починаючи з таких сфер, як економіка та право і до технічних тем, таких як аналіз даних, обґрунтованість, безпека та контроль.

Історія штучного інтелекту

Термін штучний інтелект був введений в 1956 році, але AI набув популярності сьогодні завдяки збільшенню обсягів даних, вдосконаленим алгоритмам та вдосконаленням обчислювальної потужності та сховища. Ранні дослідження ШІ у 1950-х роках досліджували такі теми, як вирішення проблем та символічні методи. У 1960-х роках Міністерство оборони США зацікавилася цим видом робіт і почало навчати комп'ютери, щоб імітувати основні людські міркування. Наприклад, Агенція прогресивних дослідницьких проєктів в галузі оборони (DARPA) завершила проєктування вуличних карт у 1970-х роках. І DARPA виготовив інтелектуальних особистих помічників у 2003 році, задовго до того, як Сірі, Алекса або Кортана були домашніми іменами. Ця рання робота проклала шлях до автоматизації та формальних міркувань, які ми бачимо сьогодні на комп'ютерах, включаючи системи підтримки прийняття рішень та системи розумного пошуку, які можуть бути розроблені для доповнення та розширення здібностей людини. Хоча голлівудські фільми та науково-фантастичні романи зображують ШІ як роботів, схожих на людей, які переймають світ, сучасний розвиток технологій ШІ не настільки страшний - або зовсім не такий розумний. Натомість AI розвивають, щоб забезпечити багато конкретних переваг у кожній галузі. Прочитайте сучасні приклади штучного інтелекту в галузі охорони здоров'я, роздрібної торгівлі тощо.

Які проблеми використання штучного інтелекту?

Штучний інтелект змінить кожну галузь, але ми повинні зрозуміти її межі. Принципове обмеження ШІ полягає в тому, що він вчиться на основі даних. Не існує іншого способу включення знань. Це означає, що будь-які неточності в даних будуть відображені в результатах. І будь-які додаткові шари прогнозування або аналізу потрібно додавати окремо. Сьогоднішні системи AI навчаються виконувати чітко визначені завдання. Система, яка грає в покер, не може грати в пасьянс або шахи. Система, яка виявляє шахрайство, не може керувати автомобілем або надавати вам юридичну пораду. Насправді система ШІ, яка виявляє шахрайство з охороною здоров'я, не може точно виявити податкові шахрайства або шахрайські вимоги. Іншими словами, ці системи дуже і дуже спеціалізовані. Вони зосереджені на одному завданні і далекі від того, щоб поводитись як люди. Так само системи самонавчання не є автономними системами. Уявлені технології AI, які ви бачите у фільмах та телебаченні, все ще є фантастикою. Але комп'ютери, які можуть досліджувати складні дані для вивчення та вдосконалення конкретних завдань, стають досить поширеними.

1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Способи збору даних

Збір даних можна вести в активному чи пасивному режимі. Пасивний режим збору інформації означає, що ми збираємо дані без можливості їх запиту. Іншими словами, це спосіб, коли запам'ятовується потрапляюча до нас інформацію без нашого спеціального на те бажання або цілеспрямованих дій по збору. Така інформація не може бути від самого початку структурована нами, оскільки нам в загальному випадку невідома структура і частота отримуваної інформації.

Активний режим збору даних визначається як режим збору, при якому ми самостійно ініціюємо запит на інформацію, в якій ми зацікавлені.

Звісно, що інформація не збирається виключно пасивно або активно. Так, при отриманні інформації в котрій ми зацікавлені в пасивному режимі, ми можемо зробити уточнюючий запит, щоб отримати додаткову інформацію. Також у відповідь на активний запит ми можемо отримати не повністю шукану для нас відповідь, але пов'язану з суб'єктом запиту.

Для активного збору даних ми повинні визначити послідовність збору та набір інформації в якій ми зацікавлені. Оскільки ми уявляємо ціль збору даних, а також їх об'єм, котрі треба зібрати, то в більшості випадків ми можемо вибудувати структуру інформації.

1.2 Контент аналіз

Аналіз контенту - це інструмент дослідження, який використовується для визначення наявності певних слів, тем чи понять у деяких даних якісних даних (тобто тексту). Використовуючи контент-аналіз, дослідники можуть кількісно оцінити та проаналізувати наявність, значення та зв'язки таких певних слів, тем чи понять. Потім дослідники можуть робити висновки про повідомлення в текстах, письменнику, аудиторії та навіть про культуру та час оточення тексту.

Опис

Джерелами даних можуть бути інтерв'ю, відкриті запитання, записки на місцях, бесіди або буквально будь-яка поява комунікативної мови (наприклад, книги, нариси, дискусії, заголовки газет, виступи, ЗМІ, історичні документи). Одне дослідження може аналізувати різні форми тексту. Для аналізу тексту за допомогою аналізу контенту текст повинен бути закодований або розбитий на категорії керованих кодів для аналізу (тобто "коди"). Після того, як текст кодується в категорії кодів, коди потім можна класифікувати на «категорії кодів», щоб ще більше узагальнити дані.

Використання контент-аналізу

- Визначте наміри, спрямованість чи тенденції спілкування людини, групи чи установи
- Охарактеризуйте ставлення та поведінкові реакції на спілкування
- Визначте психологічний або емоційний стан осіб або груп
- Виявіть міжнародні відмінності у змісті спілкування
- Розкрийте закономірності у змісті спілкування
- Попередньо протестуйте та вдосконалюйте втручання чи опитування перед запуском
- Проаналізуйте інтерв'ю фокус-груп та відкриті запитання для доповнення кількісних даних

1.2.1 Види контент-аналізу

Існує два загальних типи контент-аналізу:

концептуальний аналіз та реляційний аналіз.

- Концептуальний аналіз визначає існування та частоту понять у тексті.
- Реляційний аналіз надалі розвиває концептуальний аналіз шляхом вивчення зв'язків між поняттями в тексті. Кожен вид аналізу може призвести до різних результатів, висновків, інтерпретацій та смислів.

Концептуальний аналіз

Зазвичай люди думають про концептуальний аналіз, коли думають про аналіз контенту. У концептуальному аналізі для обстеження вибирається поняття, яке включає кількісне визначення та підрахунок його присутності. Основна мета - вивчити наявність вибраних термінів у даних. Умови можуть бути явними або неявними. Явні терміни легко визначити. Кодування неявних термінів складніше: потрібно визначити рівень імплікації та базувати судження про суб'єктивність (питання щодо надійності та обґрунтованості). Тому кодування неявних термінів передбачає використання словника чи правил контекстного перекладу або те і інше.

Щоб розпочати концептуальний аналіз контенту, спочатку визначте питання дослідження та виберіть зразок або зразки для аналізу. Далі текст повинен бути закодований в категорії керованого вмісту. Це в основному процес виборчого зниження. Зводячи текст до категорій, дослідник може зосередитись та кодувати конкретні слова чи зразки, які інформують про дослідницьке питання.

Реляційний аналіз

Реляційний аналіз починається як концептуальний аналіз, де для обстеження вибирається концепція. Однак аналіз передбачає вивчення взаємозв'язків між поняттями. Окремі поняття розглядаються як такі, що не мають притаманного їм значення, а скоріше значення є продуктом взаємозв'язків між поняттями.

Щоб розпочати аналіз реляційного контенту, спочатку визначте дослідницьке питання та виберіть зразок або зразки для аналізу. Питання дослідження повинно бути зосереджено, щоб типи понять не піддавалися

інтерпретації та могли бути узагальнені. Далі виберіть текст для аналізу. Виберіть текст для аналізу ретельно, врівноваживши достатню кількість інформації для ретельного аналізу, щоб результати не обмежувалися наявністю надто обширної інформації, щоб процес кодування став надто важким і важким, щоб забезпечити значущі та варті результати.

1.2.2 Переваги та недоліки

Переваги контент-аналізу

- Безпосередньо вивчає спілкування за допомогою тексту
- Дозволяє як якісний, так і кількісний аналіз
- Дає цінні історичні та культурні уявлення з часом
- Дозволяє близькості до даних
- Зашифрована форма тексту може бути статистично проаналізована
- Ненав'язливі засоби аналізу взаємодій
- Дає розуміння складних моделей людської думки та використання мови
- Якщо все зроблено добре, вважається відносно «точним» методом дослідження
- Аналіз вмісту - це легко зрозумілий і недорогий метод дослідження
- Більш потужний інструмент у поєднанні з іншими методами дослідження, такими як інтерв'ю, спостереження та використання архівних записів. Це дуже корисно для аналізу історичного матеріалу, особливо для документування тенденцій у часі.

Недоліки контент-аналізу

- Це може бути надзвичайно трудомістким
- Піддається підвищеній помилці, особливо коли реляційний аналіз використовується для досягнення більш високого рівня інтерпретації
- Часто позбавлений теоретичної бази або намагається занадто вільно зробити змістовні умовиводи щодо взаємозв'язків та наслідків, що містяться в дослідженні
- По суті є редуktivним, особливо коли йдеться про складні тексти
- Занадто часто схильний просто складатися з підрахунку слів
- Часто ігнорує контекст, який створив текст, а також стан речей після створення тексту

1.3 Соціальна мережа Instagram

Коли стартував Instagram?

Після декількох місяців спроб Instagram був створений та запущений 6 жовтня 2010 року. У перший день в додатку з'явилися двадцять п'ять тисяч користувачів! Але це не був надмірний успіх. Бажанням користувачів було створення візуальних (і просто оптичних СМ), де можна поділитися деякими своїми найновішими знімками, або знайти своїх старих друзів дитинства і подивитися, як вони виглядають зараз!

Зростання мережі

Якщо в січні 2013 року число активних користувачів на місяць дорівнювало 90 мільйонам, то до червня 2018 року воно збільшилось до 1 мільярду.

Той факт, що число щоденних активних користувачів Instagram дорівнює 500 мільйонам, дозволяє цій соціальній мережі вважатися популярнішою, ніж Twitter (36 млн. Активних користувачів), Snapchat (150 млн.) та Pinterest (250 млн.).

Частка дорослих інтернет-користувачів, що користуються Instagram, зросла практично на 400% з тих пір, як Pew Research Center вперше почав відстежувати цей параметр в 2012 році.

Призначена для користувача база Instagram зростає набагато швидше, ніж в цілому обсяг використання соціальних мереж у світі. Проте прогнозується, що в наступні кілька років Instagram буде рости повільніше (з 9,1% зростання в 2020 році до 4% в 2022 році).

З 2016 по 2020 рік, за прогнозами eMarketer, база Instagram стане більше на 26,9 мільйонів користувачів - це майже вдвічі більше очікуваного приросту у Twitter і набагато більше, ніж у будь-якій іншій соціальній платформи, зростання якої відслідковується.

Очікується, що в 2020 році чверть всього доходу від реклами Facebook заробить на Instagram. До 2021 року частка Instagram в цьому доході зросте до 30%.

Instagram і Snapchat є другими за популярністю додатками для обміну повідомленнями (перше місце займає Facebook Messenger) серед мілленіалів (47%).

Станом на січень 2019 року, крім облікового запису самого Instagram, найпопулярнішим є аккаунт футболіста Кріштіану Роналду, на якого підписані такі відомі люди, як Селена Гомес, Аріана Гранде і Дуейн (Rock) Джонсон.

1.3.1 Аналіз Instagram

Аудиторія та демографія

1. 72% підлітків користуються соціальною мережею Instagram.
2. Якщо у вересні 2014 року частка дорослих користувачів Instagram становила 26%, то до 2018 року вона збільшилася до 35%.
3. У 2013 році частка користувачів від 18 до 24 років (так звані young adults) дорівнювала 37%, а до 2018 збільшилася до 71%.
4. У 2018 році 60% користувачів Instagram щодня використовували платформу, в тому числі 55% користувачів від 18 до 24 років, які відвідували її по кілька разів на день. У 2016 році щоденним відвідуванням сайту могли похвалитися 51% користувачів.
5. Більше третини користувачів Instagram робили покупки в інтернеті через свої мобільні пристрої - на 70% частіше, ніж люди, які не користуються Instagram.
6. Більше половини користувачів Instagram в Латинській Америці - жінки (56%).
7. Більше 80% користувачів Instagram живуть за межами США.
8. Країни з найбільшим числом користувачів Instagram (станом на травень 2020 року): США (121 млн.), Індія (71 млн) і Бразилія (64 млн.).

Адаптація бренду

9. Компанії, які знаходяться в Instagram, до 37% від загального числа показів отримують з Instagram Stories.
10. 8 з 15 найпопулярніших брендів в Instagram належать до сфери роздрібної торгівлі.
11. Приблизно 80% всіх користувачів Instagram підписані на той чи інший бізнес-аккаунт.
12. Згідно з дослідженням, проведеним в 2018 році, найвищий рівень залученості на пост генерувала галузь вищої освіти (3,39%). Друге місце зайняли спортивні команди - 2,28%, а третє - некомерційні організації (2,14%).

Зміст постів в Instagram

13. В ході дослідження 8 мільйонів публікацій Instagram було виявлено, що ті з них, в яких домінував якийсь один колір, генерували на 17% більше «лайків», ніж ті, в яких таких кольорів було кілька.
14. У Instagram зображення з великою кількістю негативного простору генерують на 29% більше лайків, ніж з мінімальним негативним простором.
15. Чотири з п'яти найпопулярніших емодзі - це позитивні смайлики (включаючи смайл «плачу від сміху»). Якщо ви подивитесь на першу 20-ку популярних емодзі, не дивуйтеся тому, що смайли складають половину.
16. Показник залучення у постів з геотеги на 79% вище, ніж у постів без геотегов.

Постинг-стратегія в Instagram

17. У ході вивчення постинг-стратегії 5 брендів було виявлено, що середня частота розміщення постів у компаній становить 1,5 пости в день.
18. Кращий час для публікації в Instagram - четвер з 5 до 11 годин вечора.
19. Найпоширеніша частота розміщення постів у брендів становить 11-20 публікацій в місяць (це характерно майже для третини всіх компаній в соціальній мережі).
20. 90% входять до «Interbrand 100» компаній мають аккаунти в Instagram. 80% з них в тиждень публікують як мінімум одну фотографію або відео.
21. Більшість постів набирають половину від підсумкового числа коментарів за перших 19 годин. Ще 10% коментарів буде отримано протягом наступних 19 днів.
22. Для багатьох постів низький рівень активності характерний протягом декількох днів і навіть тижнів після публікації. Більшість постів брендів продовжують отримувати лайки і коментарі через добу після публікації, тільки в більш повільному темпі.
23. Незважаючи на те, що Instagram як і раніше є кращою соціальною мережею для органічної взаємодії, коефіцієнт взаємодії з кожним підписником дорівнює 2,2%, і це вдвічі менше, ніж в 2014 році.

24. Вивчення 100 найпопулярніших брендів показало, що рівень активності для посту виріс на 53% в порівнянні з аналогічним періодом минулого року.
25. При дослідженні декількох тисяч постів різних брендів було з'ясовано, що середній рівень активності становить 4,3%, а медіана - 3,5%. Це означає, що в середньому кожен 100 підписників над кожним постом здійснюють 4,3 дії (лайк або коментар). Таким чином, щоб отримати 100 лайків і коментарів до посту, бренду знадобиться приблизно 2 325 підписників.

Важлива хронологія Instagram

- 6 жовтня 2010 року: запуск соціальної медіа Instagram на IOS.
- 12 грудня 2010 року: в Instagram з'явився перший мільйон користувачів.
- 21 березня 2011 року: перший у світі InstaMeet.
- 3 серпня 2011 року: до Instagram завантаження 150 мільйонів фотографій .
- 20 вересня 2011 року: представлена версія 2.0 з чотирма новими фільтрами.
- 26 вересня 2011 року: 10 мільйонів користувачів.
- 21 листопада 2011 року: запуск першого хештег-проекту #VHP у вихідні дні.
- 3 квітня 2012 року: Instagram також запускається на android.
- 9 квітня 2012 року: Facebook купує Instagram.
- 28 червня 2012 року: Instagram розробляє фото-сторінки для Інтернету.
- 26 липня 2012 року: в Instagram є 80 мільйонів користувачів.
- 11 грудня 2012 року: версія 3.2 з новим фільтром.
- 21 грудня 2012 року: версія 3.4.1 пропонує 25 мов.
- 2 травня 2013 року: дозволяється позначати фотографії в Instagram.
- 20 червня 2013 року: представлення відео про Instagram.

12 грудня 2013 року: представлення Instagram direct.

1 грудня 2014 року: Instagram сягає 300 мільйонів.

22 жовтня 2015 року: запуск бумерангу.

15 грудня 2016 року: Instagram досягає 600 мільйонів.

22 лютого 2017 року: Instagram дозволяє надсилатикілька фотографій за повідомлення.

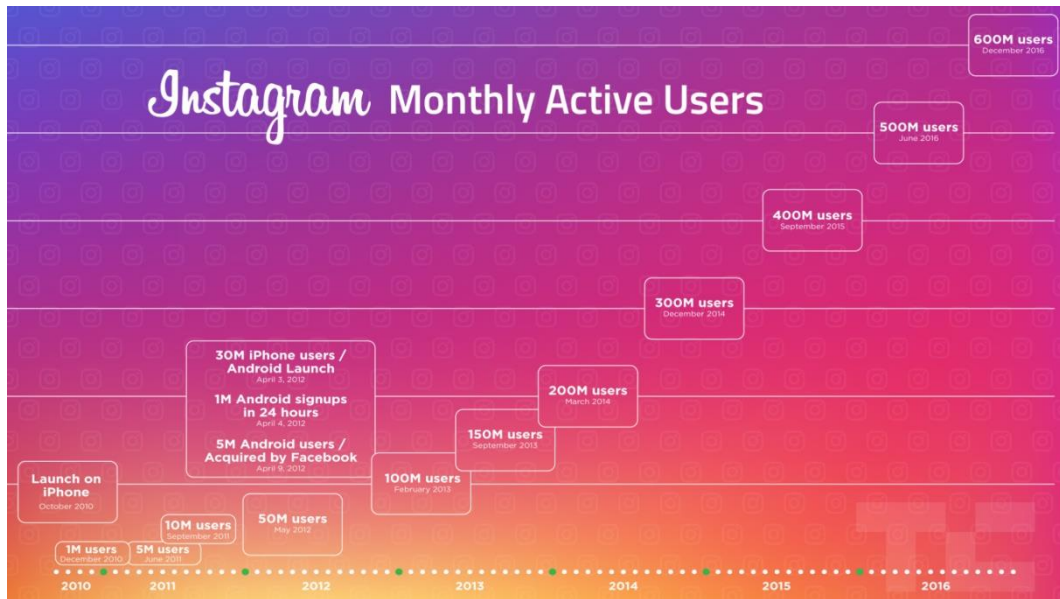


Рисунок 1.2 – Демонстрація зростання Instagram

1.3.2 Що зробило Instagram таким популярним?

Я вважаю, що є кілька причин цього успіху:

- Instagram дозволяє робити зі звичайних фотографії неординарні за допомогою використання різних фільтрів.
- Люди не хочуть читати довгі тексти, тому їм більше цікаво переглядати відео та фотографії, і це основна лінія Instagram.
- Instagram простий у використанні та безкоштовний.
- Instagram зосередився на тому, щоб робити одне, а не намагатися робити занадто багато речей.
- Соціальна мережа має безліч цікавих функцій та оновлення відповідно до щоденних потреб користувачів.
- В мережі не міститься занадто багато рекламних оголошень, нічого зайвого не спливає!

- Ви можете зробити відмінний бізнес-профіль і заробити багато грошей на цьому.

Instagram - це гігант.

Зазвичай, коли виходить нова платформа соціальних медіа, люди ігнорують її якомога довше - адже вже існує стільки різних платформ. Але через деякий час певні з них стають занадто великими, щоб їх ігнорувати. Facebook, YouTube, Twitter і навіть Pinterest досягли цієї стадії, і я думаю, що Instagram зараз для цієї галузі є, навіть більшим, ніж це було всього півроку тому.

Фінансування

Завдяки масовому зростанню користувачів в Instagram, інвестори зацікавилися. У січні 2011 року компанія почала зустрічатися із зацікавленими сторонами, і через місяць після цього соціальна медіа Instagram зібрала близько 7 мільйонів доларів у різних інвесторів, включаючи базовий капітал. Після цього компанія привернула увагу провідних компаній у соціальних мережах, twitter та Facebook.

Це дозволило Systrom та Krieger найняти більше людей.

Інтереси з твіттера

Дорсі знав Systrom ще від стажування в Одео, і він дуже зацікавився компанією. Так, у березні 2011 року він запропонував близько 500 мільйонів доларів запасів, як повідомляється. Systrom все ж таки відмовились; вони вважали за краще, щоб Instagram залишався незалежною компанією.

До березня 2012 року Instagram досяг 27 мільйонів користувачів, і він був випущений для телефонів на платформі Android у квітні. Його завантажили більше ніж один мільйон разів, за перший день. Компанія отримала новий етап фінансування на суму 500 мільйонів доларів.

Інтереси від Facebook

У квітні 2012 року Марк Цукерберг (засновник Facebook) зробив пропозицію придбати Instagram за 1 мільярд доларів готівкою та запасами, за умови, що компанією залишатиметься самостійно управляючою. Незабаром після цього Facebook придбав компанію за 1 мільярд доларів готівкою та запасами.

1.3.3 Що таке особистий аккаунт в Instagram?

Володіння особистим аккаунтом в Instagram - це не щось дивне. Ви можете мати його навіть без створення власного веб-сайту та спеціального домену. Instagram відкрив вікно можливостей для тих, хто бажає продовжити кар'єру блогера у своїй улюбленій ніші, вести блог компанії та рекламувати її або просто ж вести особисту сторінку.

Кожен користувач має можливість створити бізнес-акаунт, як блогер. Потрібно перейти до розділу меню "Редагувати профіль" та обрати категорію "Особистий блог". Блогери Instagram ставляться до своєї сторінки точно так, як до свого персонального блогу.

Потрібно заповнити біо:

- Користувачу потрібно описати, хто він такий, чим займається і як може допомогти своїми цілям.
- Якщо у користувача є інші облікові записи соціальних мереж, потрібно перейти за посиланням у розділ меню «Веб-сайт». Там є URL-адреса, яку потрібно заповнити.
- Прикарасити свій біо, використовуючи символи, емодзі або смайлики.
- Змінити шрифт свого біо з нудного на привабливий. Однак потрібно заздалегідь перевірити читабельність.

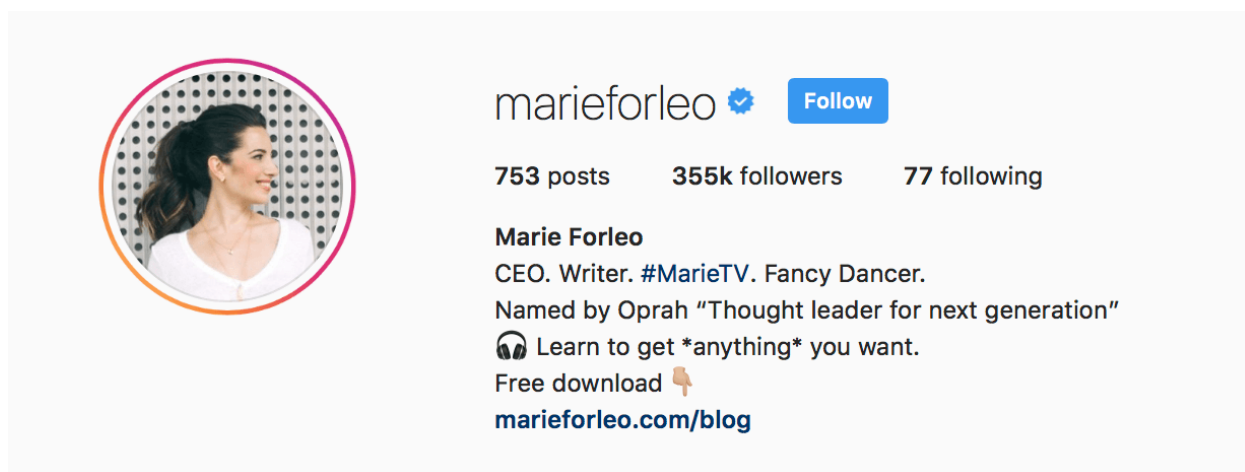


Рисунок 1.3 – Приклад біо у користувача Instagram

- Обрати привабливу фотографію для вашого профілю. Перше, що може привернути увагу відвідувачів - це зображення профілю та біо опис.
- Обрати палітру кольорів та тему для оформлення свого профілю в Instagram. Це надасть змогу аккаунту виглядати більш професійно. Ви можете дотримуватися цього дизайну на інших ваших платформах соціальних медіа.
- Корисно використовувати хештеги. До речі, треба визначитися з деякими фірмовими хештегами, які стосуються лише вас. Одного-двох зазвичай вистачить.



Рисунок 1.4 – Приклад хештегів у користувача Instagram

- Розділити свій рекламний та нерекламний вміст, щоб переконатися, що ваш вміст не нудний. Будьте в курсі щоденних новин і поширюйте їх, коли це можливо, щоб ефективно залучати своїх підписників.

- Швидко реагувати на повідомлення чи коментарі. Проявляйте подяку, коли користувачі реагують на ваш вміст або пропонують ідеї, як покращити вашу сторінку.
- Не варто недооцінювати силу історій. У наші дні люди більше дивляться саме історії, ніж дописи у стрічці. Наприклад, ви можете оголосити нову публікацію у своїй історії чи продемонструвати відгуки.

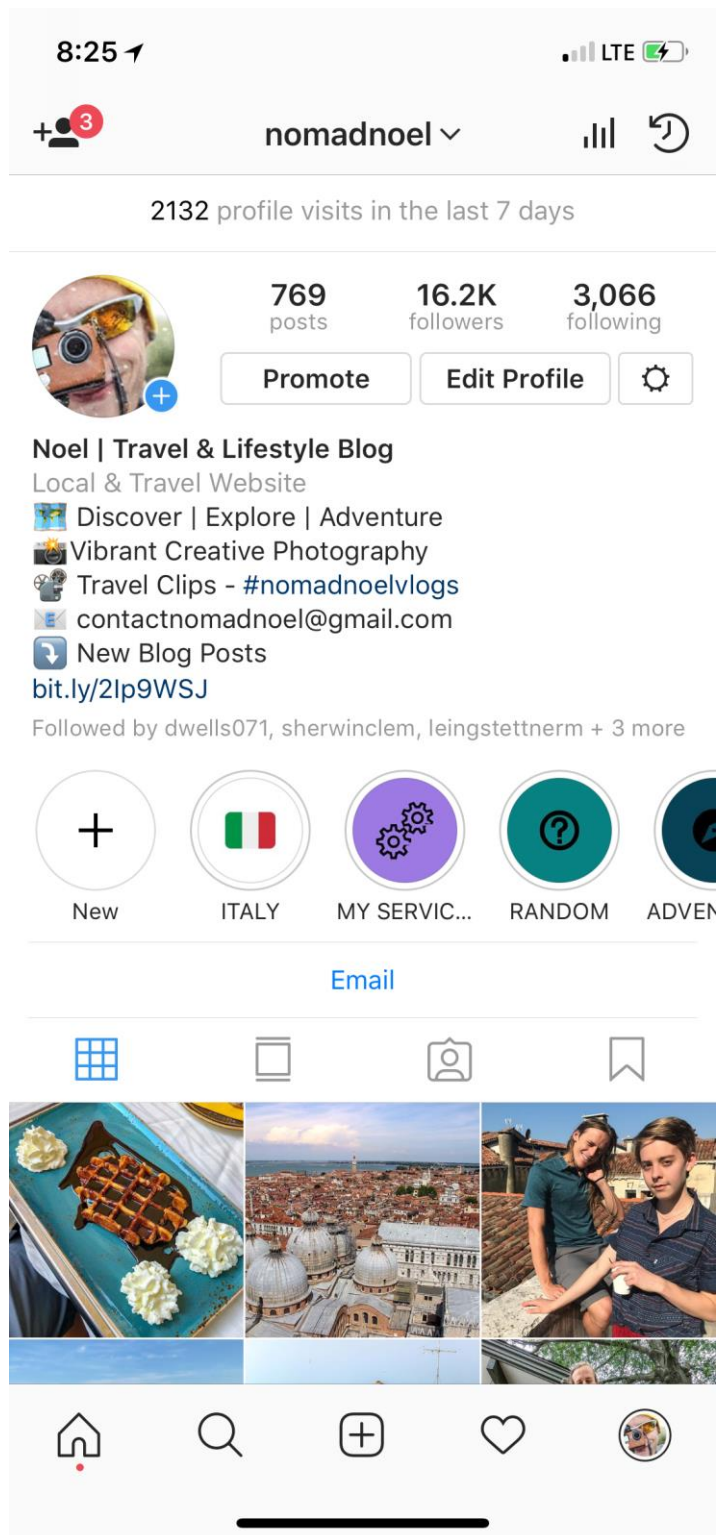


Рисунок 1.5 – Приклад наповненого контентом аккаунта у користувача Instagram

1.3.4 Використання штучного інтелекту в Instagram

Завдяки використанню хештегів користувачі Instagram можуть знайти фотографії для певної діяльності, теми чи події або детальніше дізнатися про певні події або місця у всьому світі. Інструменти пошуку за допомогою хештегів допомагають користувачам Instagram виявляти цікаві речі серед мільйонів завантажених зображень.

Щоб користувачі знаходили цінність на платформі, Instagram важливо показати їм, те що їм подобається. Зі збільшенням кількості контенту пошук вмісту, який кожен користувач вважатиме релевантним, стає експоненціально складнішим. Коли Instagram змінив свій канал із зворотного хронологічного порядку на показ публікацій, які, як вони вважають, користувачам подобаються та вони ними діляться, алгоритми машинного навчання були поставлені на роботу, щоб допомогти сортувати інформацію та краще з часом вивчити те, що найбільше цінується та актуально для кожного користувача, щоб створити персоналізований канал.

Instagram використовує штучний інтелект для боротьби зі спамом.

Фільтр спаму може видаляти підроблені повідомлення з облікових записів, написаних дев'ятьма мовами, включаючи англійську, китайську, арабську та інші. Після виявлення повідомлень вони автоматично видаляються. Instagram використовує алгоритм штучного інтелекту Facebook DeepText, який здатний зрозуміти контекст повідомлення майже так само добре, як і люди.

Під час опитування, випадково обраних користувачів, 42% з понад 10 000 молодих людей у всьому світі у віці від 12 до 25 років повідомили, що Instagram був платформою, де вони найбільше зазнавали знущань. З цією прикрою відмінністю, що має найбільшу проблему в кібер-насилстві будь-якого сайту соціальних медіа, Instagram стали першими, хто використовував машинне навчання для автоматичного видалення образливих дописів, тоді як Facebook та Twitter покладаються на користувачів, щоб повідомити про нецензурну мову.

В одному з досліджень 100 мільйонів фотографій в Instagram були використані для вивчення глобальних моделей одягу. Те, що було б неможливим для перегляду даних, стає все більш можливим завдяки машинному навчанню. Ця робота показала потенціал машинного навчання, щоб допомогти отримати уявлення під час вивчення людей та соціальних,

економічних та культурних факторів у всьому світі. Це дослідження показало потужність великих даних і таких технологій, як комп'ютерний зір, алгоритми автоматизованого аналізу та машинне навчання для комбінування за допомогою величезних наборів даних, які створюються сайтами соціальних медіа, щоб отримати розуміння про навколишній світ.

Для того, щоб зробити дані, які Instagram збирає цінними, система повинна витягнути з нього інформацію про клієнтів. Оцінюючи налаштування пошуку та розуміння зацікавленості користувачів, Instagram може продавати рекламу компаніям, які хочуть охопити конкретний профіль клієнта і які можуть бути найбільш зацікавлені отримати певне маркетингове повідомлення.

Пошук потрібної цільової аудиторії в соціальних мережах часто викликає багато складнощів у невідповідної людини. ІІІ може стати провідником у світі інтернет-реклами, допомагаючи заощадити багато часу і сил у користувача. За допомогою створеної системи можна в пару кліків налаштувати вимоги до шуканої цільової аудиторії і в найкоротші терміни залучити потенційних підписників.

Користувач має можливість шукати профілі, хештеги та місця, а Instagram поверне топ-100 публікацій.

Коли ви відкриєте будь-яку загальнодоступну сторінку Instagram, яка містить публікації (наприклад, профіль, хештег або місце), Instagram поверне HTML-сторінку з попередньо завантаженими першими публікаціями (можливо, використовуючи React-рендерінг на стороні сервера). Потім, коли ви прокрутите сторінку вниз, Instagram продовжить завантажувати більше публікацій, використовуючи запит XHR, до кінцевої точки GraphQL в Instagram. Кінцева точка захищена токеном, тому реально отримати доступ до неї неможливо, і нам потрібно нескінченно прокручувати сторінку. Однак можливо автоматизувати нескінченне прокручування.

1.4 Обмеження Instagram, які вплинули на реалізацію ПП

Соціальна мережа Instagram активно бореться проти використання занадто великої кількості автоматизованого ПЗ на своїх ресурсах.

Найбільш очевидна ознака отримання тимчасового блоку від Instagram - це намагання виростити за один раз значну кількість лайків та підписників. Це може статися в обох випадках, якщо ви занадто багато використовували сторонні інструменти просування або виконували вручну дії, не вивчаючи профілів користувачів.

Ліміти і обмеження в Інстаграм - це певні правила, яких слід дотримуватися, щоб аккаунт не потрапив під блокування.

Пам'ятайте: Instagram активно бореться з акаунтами, які просуваються забороненими методами.

Тому, щоб уникнути санкцій - довіряйте тільки актуальним даними.

1.4.1 Дослідження лімітів Instagram

Для точних даних по лімітам я провів незалежний експеримент. Так як в інтернеті повно неактуальною інформації щодо лімітів, я вирішив особисто перевірити, скільки лайків можна ставити та скільки можна робити підписок без шкоди для аккаунта.

Скільки в день можна підписуватися в Instagram?

Насамперед дізнаємося, на скількох користувачів можна підписуватися максимум на годину / день (в залежності від віку аккаунта):

- Новий акаунт: на годину від 10 до 20, в день від 125 до 175;
- Молодий аккаунт (від 3 місяців до 1 року): на годину від 15 до 30, в день від 200 до 300;
- Старий аккаунт (від 1 року): на годину від 20 до 30, в день від 250 до 300.

В ліміти входить сукупність підписок і відписок. Наприклад, новий аккаунт може підписатися на 100 користувачів і відписатися від 75 (потрібно

вкластися в максимальну межу - 175). Або просто підписатися на 175 користувачів, але відписуватися вже можна тільки на наступний день.

Рекомендації

Новий акаунт.

Після створення рекомендую додати кілька публікацій (фото, відео) і обов'язково аватарку(фото профілю).

У першу годину можете підписатися на 10 користувачів. У день не більше, ніж на 125.

У наступні дні можете спокійно підписуватися на 10 до 20 акаунтів на годину і 175 в день.

Пам'ятайте, що Інстаграм стежить за діями і не важливо, підписуєтеся ви або відписуються. Так що стежте за тим, щоб не вийти із загального ліміту.

Молодий аккаунт.

Щогодини ви можете сміливо підписуватися на 25 користувачів, але не більше. У день краще не перевищувати планку в 200 акаунтів.

Старий аккаунт.

Ви надійний користувач і без проблем можете підписуватися на 30 акаунтів на годину і 300 в день.

Скільки в день можна ставити лайків в Instagram?

Тепер поговоримо про обмеження для лайків. Тут ліміти вище, але перевищувати їх теж не варто.

- Новий акаунт: на годину від 10 до 20, в день від 400 до 500;
- Молодий аккаунт (від 3 місяців до 1 року): на годину від 20 до 40, в день від 500 до 1000;
- Старий аккаунт (від 1 року): на годину від 30 до 50, в день від 800 до 1200.

В ліміти входять всі дії (неважливо, ставите або прибираєте лайк). Наприклад, новий акаунт може в день поставити 400 лайків і прибрати 150 (або поставити 550 лайків, але прибрати їх в цей день не варто - буде перевищення).

Рекомендації

Новий акаунт.

Новому аккаунту найкраще почати з додавання кількох публікацій (фото або відео) і аватарки(фото профілю).

В першу годину можете поставити 15 лайків.

Але в день не більше 450.

А вже в наступні дні можете спокійно ставити до 25 лайків на годину і не більше 550 в день.

Головне не вийти з ліміту. Причому в ліміт входять лайкі, які ви ставите і забираєте.

Молодий аккаунт.

В першу годину ви можете ставити до 40 лайків, але не більше.

У день краще не ставити понад 800 лайків.

Старий акаунти.

Старим акаунтів можна спокійно ставити до 50 лайків на годину і до 1200 в день.

Чи можна поєднувати дії, слідуючи лімітами? Наприклад, в один день підписуватися по максимуму, ставити лайки і коментувати?

По максимуму не можна. Потрібно скористатися формулою: мінімальна кількість лімітів поділити на два.

Наприклад, молодому аккаунту в день можна підписатися на 100 користувачів, поставити 250 лайків

1.4.2 Які санкції можуть бути застосовані за порушення?

Тут тільки два варіанти:

- Тіньовий бан - тимчасове блокування певних дій (підписка на інших людей, публікація фото, історій і т.д.)
- Блокування аккаунта - вічний бан без можливості відновлення.

Висновки: Інстаграм нещадно банить акаунти за підозрілу активність і постійно покращує свої алгоритми. Щоб мінімізувати ризик потрапляння під блокування необхідно чітко дотримуватися ліміти і обмеження. Перед просуванням обов'язково шукайте актуальну інформацію.

Якщо дотримуватися дані ліміти та рекомендації, то ніяка блокування вам не страшна (списування підписників не буде).

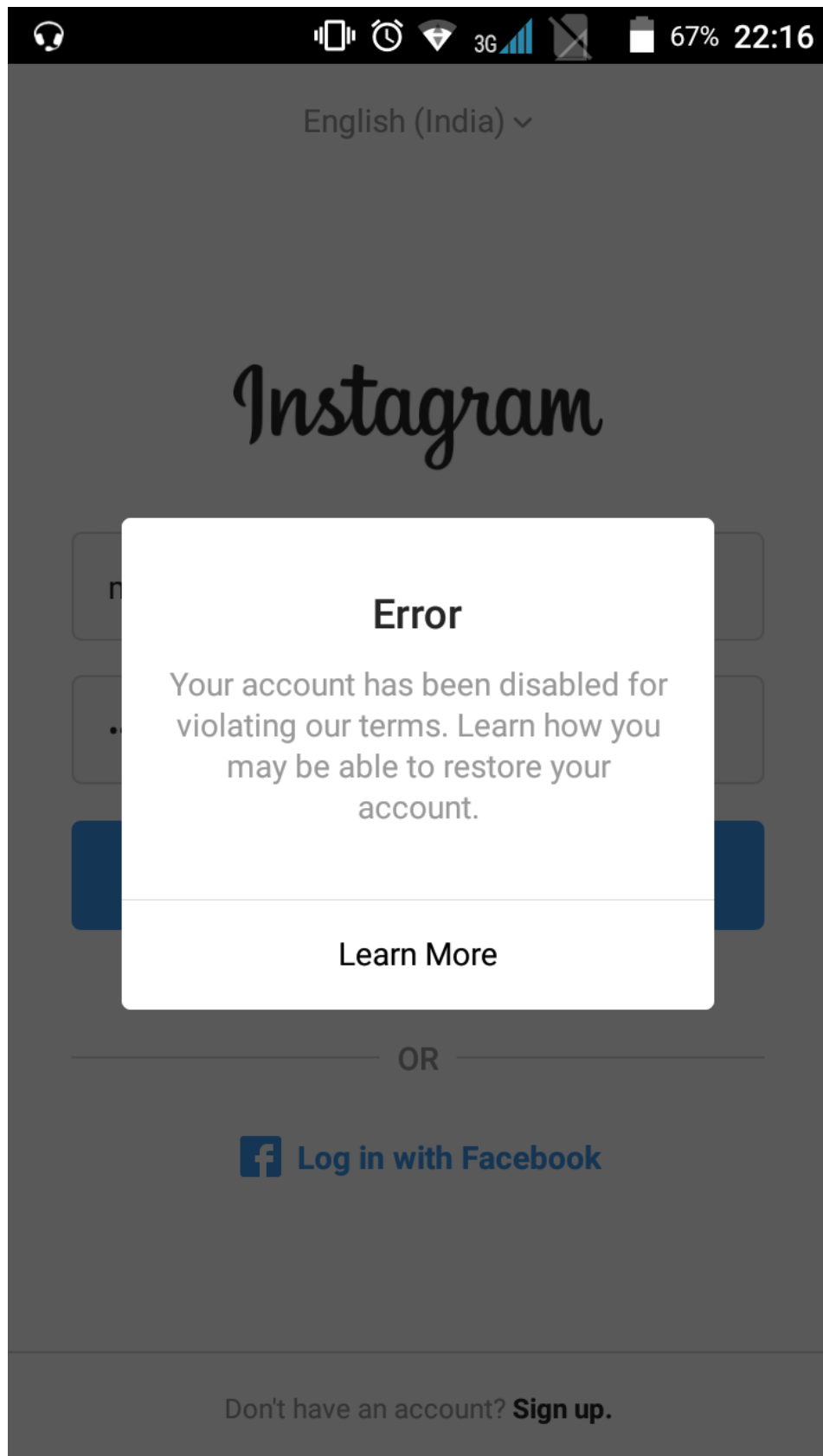


Рисунок 1.7 – Приклад тіньового бану користувача Instagram

1.5 Навіщо збирати дані Instagram-користувачів?

Instagram на сьогоднішній день має величезну кількість користувачів в усьому світі. Збираючи дані про аккаунти, користувач з легкістю може підібрати потрібно ЦА саме для своїх цілей, також платформа відображає поточні тенденції на ринку, що буде слугувати безумовним плюсом для користувача. Зараз Instagram має високий маркетинговий потенціал, що допоможе у веденні бізнесу у мережі. Недарма, експерти вважають цю соціальну мережу найкращим місцем для збору даних для аналізу настроїв

Як збирати дані з Instagram?

Маючи понад 1 мільярд користувачів, Instagram, безумовно, є платформою соціальних медіа, на якій слід зосередити увагу незалежно від того, чи є ви B2C або B2B. Використовувати Instagram для того, щоб залишатися на зв'язку зі своїми клієнтами, необхідно, але потрібно зробити ще багато, використовуючи дані, доступні в Instagram. Витяг даних з Instagram за допомогою парсеру відкриває багато нових можливостей, як для власника бізнесу, так і для блогера.

Instagram Audiences Top Interests

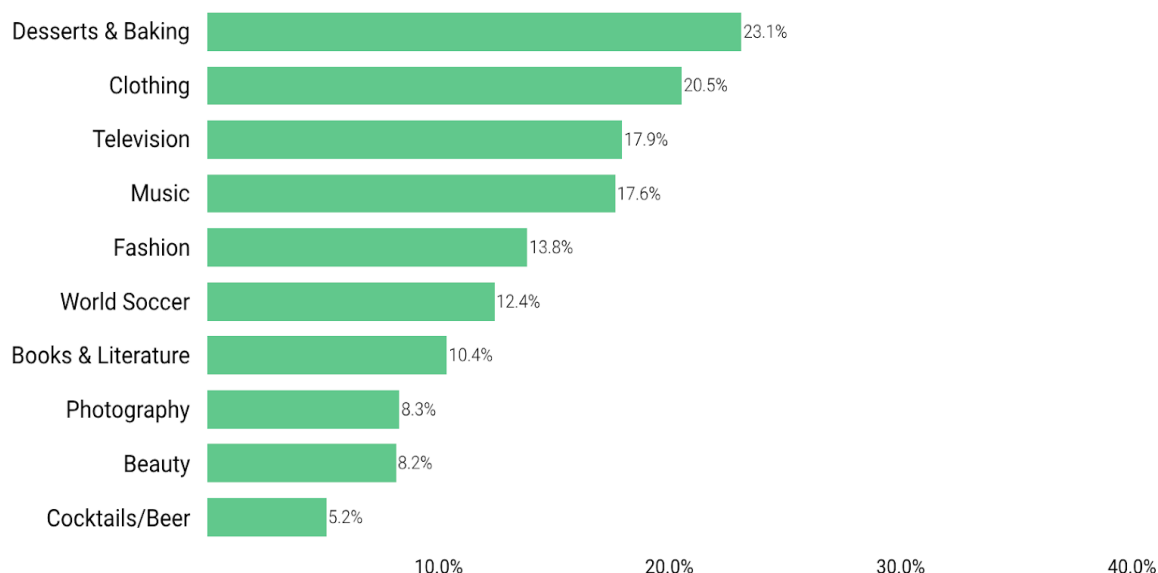


Рисунок 1.8 – Збір даних мережею Instagram про інтереси своїх користувачів

1.5.1 Активний автоматизований збір даних

У сучасних реаліях система активного автоматизованого збору даних цільової аудиторії в Instagram застосовується для залучення активних (теплих) користувачів в різноманітні аккаунти. Найчастіше користуються подібними сервісами власники аккаунтів в Instagram, так як це один з найбільших ринків збуту будь-якого виду продукції: від наклейок, до великої побутової техніки. Успішність же просування залежить від наступних факторів:

1. Аудиторія. Цілеспрямований підбір ЦА дозволяє зробити публікації якісними, тому як вони будуть показуватися реально зацікавленим особам;
2. Правильне ранжування в стрічці Інстаграма. По-справжньому якісні і цікаві для користувачів публікації ранжуються вище в новинних стрічках. Самі позиції визначаються виходячи з інтересів підписників;
3. Легкість пошуку. Можливість знайти потрібний пост серед тисячі інших - це один з найпривабливіших факторів, який в значній мірі підвищує зручність і комфорт користувачів. Для цього важливо правильно розставляти хештеги з урахуванням тематики і спрямованості публікації.
4. Найефективнішим же помічником у такому випадку вважається парсинг. Рекламодавці з задоволенням приходять на майданчик, а функціонал програмного забезпечення дозволяє максимально швидко домогтися їхнього розташування

Дана інформація дозволяє фахівцям правильно налаштувати інстаграм-аккаунт, щоб пости були ефективні і залучали підписників або клієнтів.

2. ЗАСОБИ РОЗРОБКИ

Далеко не всі мови програмування добре підходять для розробки програм, що реалізують методи машинного навчання для збору даних. Адже деякі з мов програмування мають велику кількість бібліотек, модулів та допоміжних засобів, що дуже спрощують розробку.

На рисунку 1.9 наведено статистику популярності мов програмування серед роботодавців.

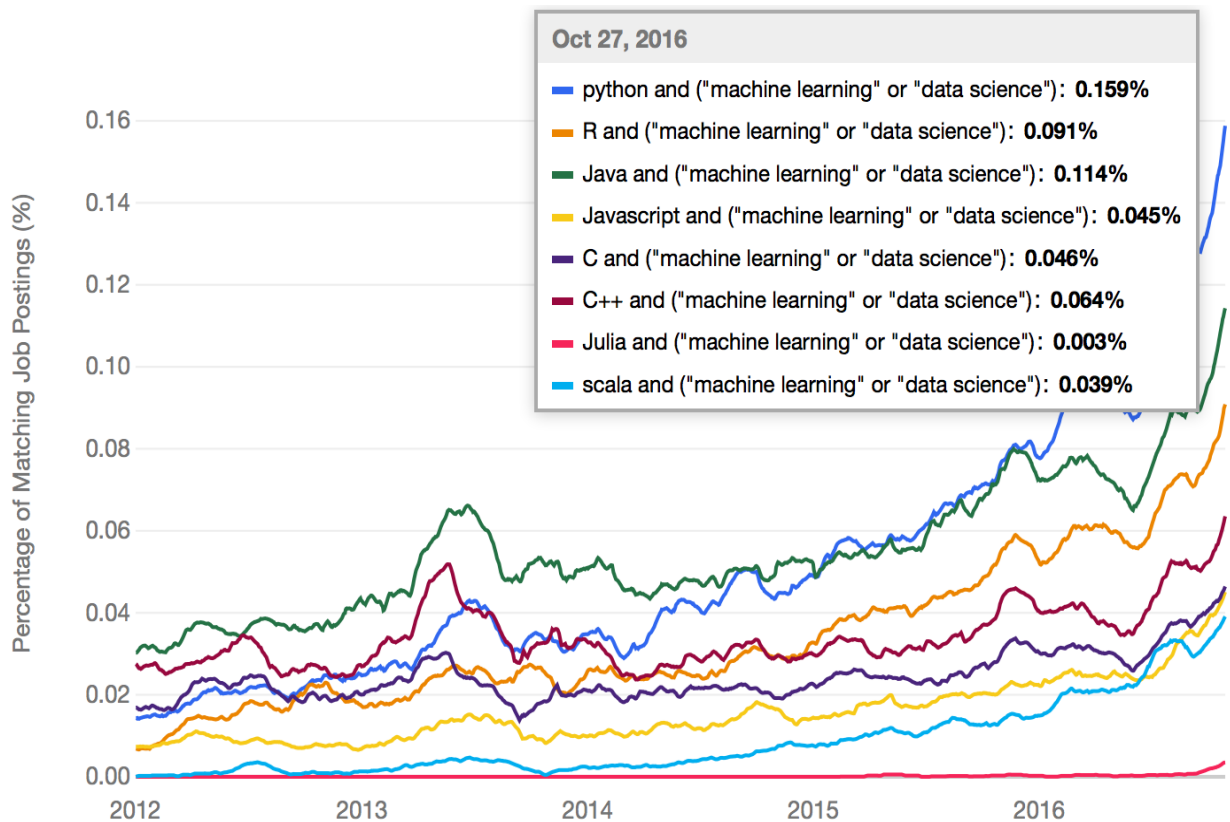


Рисунок 1.9 – Найбільш популярні мови програмування у сфері машинного навчання

Також до розгляду мов та засобів програмування, що використовуються у машинному навчанні можна додати середовища, що не так популярні серед роботодавців, але використовуються у академічній сфері. Це такі середовища, як Matlab, LabVIEW. Також, сьогодні важливе місце серед засобів розробки програм у сфері збору даних займає Selenium.

2.1 Python

Вперше розроблений наприкінці 1980-х років Гвідо ван Россумом, Python просунувся як мова програмування з відкритим кодом, керуючи громадським обговоренням через пропозиції щодо розширення програми Python (PEPs).

Наразі Python є третьою основною версією та регулярно оновлюється.

У технічному плані Python - це об'єктно-орієнтована мова програмування високого рівня з інтегрованою динамічною семантикою, насамперед для розробки інтернет-додатків. Це надзвичайно привабливо у сфері швидкого розвитку програмного забезпечення, оскільки пропонує динамічні варіанти введення тексту та динамічні функції прив'язки.

Python порівняно простий, тому його легко вивчити, оскільки він вимагає унікального синтаксису, який зосереджений на читанні. Розробники можуть читати та перекладати код Python набагато простіше, ніж інші мови. У свою чергу, це зменшує витрати на підтримку та розробку програми, оскільки дозволяє командам працювати спільно без значних бар'єрів у спілкуванні та досвіді.

Крім того, Python підтримує використання модулів і пакетів, а це означає, що програми можуть бути спроектовані в модульному стилі, а код можна використовувати повторно в різних проектах. Розробивши потрібний вам модуль або пакет, його можна масштабувати для використання в інших проектах, і їх легко імпортувати чи експортувати.

Однією з найбільш перспективних переваг Python є те, що і стандартна бібліотека, і інтерпретатор доступні безкоштовно, як у двійковій, так і у вихідній формі. Ексклюзивності також немає, оскільки Python та всі необхідні інструменти доступні на всіх основних платформах. Тому це привабливий варіант для розробників, які не хочуть турбуватися про високі витрати на розробку.

Python - це мова програмування, яка використовується для розробки програмного забезпечення в Інтернеті та у формі додатків, включаючи мобільний. Навчитися досить просто, а необхідні інструменти доступні для всіх безкоштовно.

Python на сьогоднішній день є стандартом у сфері машинного навчання. Ця мова програмування має безліч бібліотек, що спрощують розробку програм у сфері штучного інтелекту.

Python є повноцінною мовою програмування, і багато організацій використовують його у своїх виробничих системах. Мова Python це високорівнева мова програмування, що має простий синтаксис (рисунок 1.10) та низький поріг входження й вивчення, що дозволяє науковцям, що мало пов'язані з програмуванням, використовувати її у своїх дослідженнях.

Дана мова програмування через свою високорівневість є досить повільною, однак усі найбільш затратні операції машинного навчання, такі як алгоритми навчання моделей, що використовуються у Python, реалізовані на низькорівневих мовах програмування, як правило, на мові програмування С. Тож, дана мова програмування поєднує у собі зручність роботи високорівневих мов програмування, швидкість низькорівневих, а також має за плечима десятки років розробок бібліотек для реалізації алгоритмів машинного навчання.

Python script

```
1 def azureml_main(expr_as_frame):
2     import pandas as pd
3     expr = expr_as_frame.iat[0,0]
4     result = pd.DataFrame({'Expr': [expr], \
5                             'Result': [eval(expr)]})
6     return result,
```

Рисунок 1.10 – Приклад програми, написаної мовою Python

Переваги мови програмування Python:

- безкоштовість;
- синтаксис програмування. Мову програмування Python було створено схожою на природню мову, яку можна легко читати, в той час як такі мови програмування, як Matlab, орієнтовані здебільшого на вирішення математичних проблем, а не на легкість читання коду;
- потужність. Python поставляється з великими стандартними бібліотеками і має потужні типи даних, такі як списки, масиви та словники;
- простори імен. Python працює з модулями, які потрібно імпортувати для використання. Використання просторів імен дає структуру програмі і залишає її чистою та зрозумілою. У Python все є об'єктом, тому кожен об'єкт має власний простір імен. Це одна з причин, через яку код, написаний на мові Python, легко піддається відладці;
- відладка. Це впливає з об'єктно-орієнтованого характеру Python. Оскільки програма має чітку структуру, відлака проста. Приватні змінні існують лише умовно, тому розробник може отримати доступ до будь-якої частини програми, включаючи деякі внутрішні компоненти Python;
- портативність. Оскільки Python є цілком безкоштовним, код, написаний данною мовою програмування, може працювати скрізь. Крім того, він працює на операційних системах Windows, Linux і OS X;
- широкі інструменти GUI. З Python можна створити інтерфейс для програми, яка виглядає і працює добре. Програміст має вибрати будь-який з основних інструментів GUI, таких як Wx або Qt.

2.2 Selenium

Selenium автоматизація браузера в Python

Selenium - це безкоштовна і відкрита бібліотека для автоматизованого тестування веб-додатків.

Що таке Selenium?

Бібліотека спочатку була створена в 2004 році Джейсоном Хаггінс, до якого згодом приєдналися інші його колеги з компанії ThoughtWorks. Selenium підтримує всі основні браузери, і тести можуть бути написані на багатьох мовах програмування, а також працювати на платформах Windows, Linux і Macintosh.

Selenium 2 об'єднується з інших тестових фреймворком WebDriver (саме тому ви бачите 'seleniumwebdriver') на чолі з Саймоном Стюартом в Google. Сьогодні Саймон працює в FaceBook. Selenium 2.0 був випущений в липні 2011 року.

2.2.1 Мова програмування для Selenium

Тести можуть бути написані в декількох мовах програмування, таких як Java, C #, Python, JavaScript і Ruby. Тестування програмного забезпечення полягає в тому, чи відповідає робота програміста вимогам замовника. У певному сенсі, тестери представляють самих клієнтів. Тестери повинні мати більше значення при визначенні тестового синтаксису.

Модуль веб-браузера надає інтерфейс високого рівня, що дозволяє відображати веб-документи для користувачів. За більшості обставин просто виклик функції open () з цього модуля зробить правильно.

Під Unix графічні браузери є кращими від X11, але текстові браузери будуть використовуватися, якщо графічні браузери недоступні або дисплей X11 недоступний.

Якщо використовуються браузери в текстовому режимі, процес виклику блокується, поки користувач не вийде з браузера. Якщо змінна середовища BROWSER існує, вона інтерпретується як перегляд списку браузерів за замовчуванням на платформі, як список браузерів, розділених os.pathsep, щоб спробувати в порядку. Коли значення частини списку містить рядок, то вона інтерпретується як командний рядок браузера в прямому значенні, що використовується з URL-адресою аргументу, заміщеною

якщо частина не містить, вона просто інтерпретується як ім'я браузера для запуску.

Для платформ, які не є Unix-подібними, або коли на Unix доступний віддалений браузер, процес контролю не чекатиме завершення роботи браузера, але дозволить віддаленому браузеру підтримувати власні вікна на дисплеї. Якщо віддалені браузери недоступні в Unix, процес контролю запустить новий браузер і зачекає.

Веб-браузер сценарію може використовуватися як інтерфейс командного рядка для модуля. Він приймає URL як аргумент. Він приймає такі необов'язкові параметри:

- n відкриває URL, якщо це можливо, у новому вікні браузера;
- t відкриє URL-адресу на новій сторінці браузера ("вкладка").

Варіанти, природно, взаємовиключні.

Після того, як ви виконали розділ передумов, ви готові розпочати свій перший тест у Selenium мовою програмування Python! Спочатку імпортуйте класи веб-драйвера та ключів із Selenium.

```
from selenium import webdriver
```

Далі створіть примірник Chrome із контуром драйвера, який ви завантажили через веб-сайти відповідного браузера. У цьому прикладі ми припускаємо, що драйвер знаходиться в тому ж каталозі, що і сценарій Python, який ви будете виконувати.

```
driver = webdriver.Chrome('./chromedriver')
```

Якщо ви тестуєте на своїй локальній машині, це відкриє примірник Chrome локально. Ця команда дозволяє виконувати на ній тести, поки ви не скористаетесь методом `.close()` для припинення з'єднання з браузером.

Далі, використовуйте метод `.get()` драйвера для завантаження веб-сайту. Ви також можете завантажити місцевий веб-сайт для розробки, оскільки цей процес еквівалентний відкриттю вікна Chrome на локальній машині, введенні URL-адреси та натисканні клавіші Enter. Метод `.get()` не тільки починає завантажувати веб-сайт, але й чекає його повного відображення, перш ніж перейти до наступного кроку.

```
driver.get("https://instagram.com")
```

```
*PyDevModule ⓘ
1 from selenium import webdriver
2 from selenium.webdriver.common.keys import Keys
3 user_name = " "
4 password = " "
5 driver = webdriver.Firefox()
6 driver.get("https://www.facebook.com")
7 element = driver.find_element_by_id("email")
8 element.send_keys(user_name)
9 element = driver.find_element_by_id("pass")
10 element.send_keys(password)
11 element.send_keys(Keys.RETURN)
12 element.close()
13 |
```

Рисунок 1.11 – Приклад використання Selenium

3. ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

3.1 Робочий процес ПЗ

В першу чергу користувач повинен авторизувати свій аккаунт Instagram в програмному забезпеченні.

Процес роботи включає в себе збір відомостей про потенційних підписників або покупців, а саме:

кількість постів.

кількість підписок.

інформацію про тип аккаунта приватний чи ні.

дату останнього поста.

наявність фото профіля.

посилання в ВІО.

потрібну кількість аккаунтів для бази.

При цьому серед найефективніших методів залучення користувачів виділяються наступні:

Масфоловінг - метод масової підписки на людей за певними критеріями. Здійснюється підписка на людину -> йому приходить відповідне повідомлення -> він переходить на вашу сторінку і підписується в тому випадку, якщо профіль видався цікавим.

Маслайкінг - метод залучення аудиторії за допомогою проставлення лайків на пости потенційного клієнта. Проставляється лайк до однієї з публікацій -> власнику приходить повідомлення -> він переходить продивитися профіль і підписується, якщо йому близька або цікава тематика аккаунта.

При цьому працювати з парсером просто, зручно і ефективно, а сама схема включає в себе 5 основних етапів і виглядає наступним чином:

Вказуються ввідні дані для підбору аудиторії.

Вибирається джерело для пошуку потенційних клієнтів або передплатників.

Виходячи із заданих параметрів формується база користувачів із зазначенням посилання на них.

Результати експортуються в файл формату txt, який далі використовується у всіх способах просування аккаунта.(рис. 1.12)



```
pers_list.txt
1 https://www.instagram.com/_prashil_15135/
2 https://www.instagram.com/_nihad_b1vk_/
3 https://www.instagram.com/ingaedemova/
4 https://www.instagram.com/mithun_sai_m/
5 https://www.instagram.com/peroladoamaral/
6 https://www.instagram.com/edde_indonesia/
7 https://www.instagram.com/awaaa_njaay/
8 https://www.instagram.com/denelfanny/
9 https://www.instagram.com/bhrh.bgbny/
10 https://www.instagram.com/ypolepole/
11 https://www.instagram.com/nathaliavb12/
12 https://www.instagram.com/alisaiedi83/
13 https://www.instagram.com/bossofillegal/
14 https://www.instagram.com/imsaurabh_prasad/
15 https://www.instagram.com/ayouta_beauty15/
16 https://www.instagram.com/walterquintana/
17 https://www.instagram.com/connyliseth/
18 https://www.instagram.com/dxnivg/
19 https://www.instagram.com/markk8688/
20 https://www.instagram.com/sinhalpvika/
21 https://www.instagram.com/sina011166/
22 https://www.instagram.com/_tiktoks.2020/
23 https://www.instagram.com/onuro112135/
24 https://www.instagram.com/ommran.m/
25 https://www.instagram.com/prokopyev.egor/
```

Рисунок 1.12 – Зібрана база ЦА

3.1.1 Фільтрування бази ЦА

Для того, щоб база інстаграм акаунтів була якісною, потрібно правильно налаштувати фільтр. Доступні такі параметри:

Кількість підписок.

Кількість публікацій.

Виняток приватних профілів.

Дата останнього поста.

Наявність аватарки.

Небажана реклама в ВІО.

Ця функція вважається однією з найбільш корисних в парсера. Вона дозволяє швидко прибрати непотрібні сторінки користувачів і заощадити кошти в разі, якщо метою збору бази є реклама.

База зачищається від:

Ботів.

Профіль, які довгий час не використовуються.

Приватних аккаунтів.

Вивантаження бази здійснюється на комп'ютер в форматі файлу txt.(рис. 1.13).

```
≡ pers_new_list.txt
1  https://www.instagram.com/imsaurabh_prasad/
2  https://www.instagram.com/ayouta_beauty15/
3  https://www.instagram.com/bhrh.bgbny/
4  https://www.instagram.com/ypolepole/
5  https://www.instagram.com/ommran.m/
6  https://www.instagram.com/omer.mermer.5095/
7  https://www.instagram.com/atshgoosh/
8  https://www.instagram.com/ayoub__vartolu/
9  https://www.instagram.com/sania.kharal1/
10 https://www.instagram.com/aliciaaszostakiewicz/
11 https://www.instagram.com/oavzcsuzv/
12 https://www.instagram.com/wala12493/
13 https://www.instagram.com/celinadias2020/
14 https://www.instagram.com/tanz_tabriiiz/
15 https://www.instagram.com/photograghyby/
```

Рисунок 1.13 – Відфільтрована база ЦА

Далі на рисунках буде наведено приклади аккаунтів, які були видалені з бази ЦА.

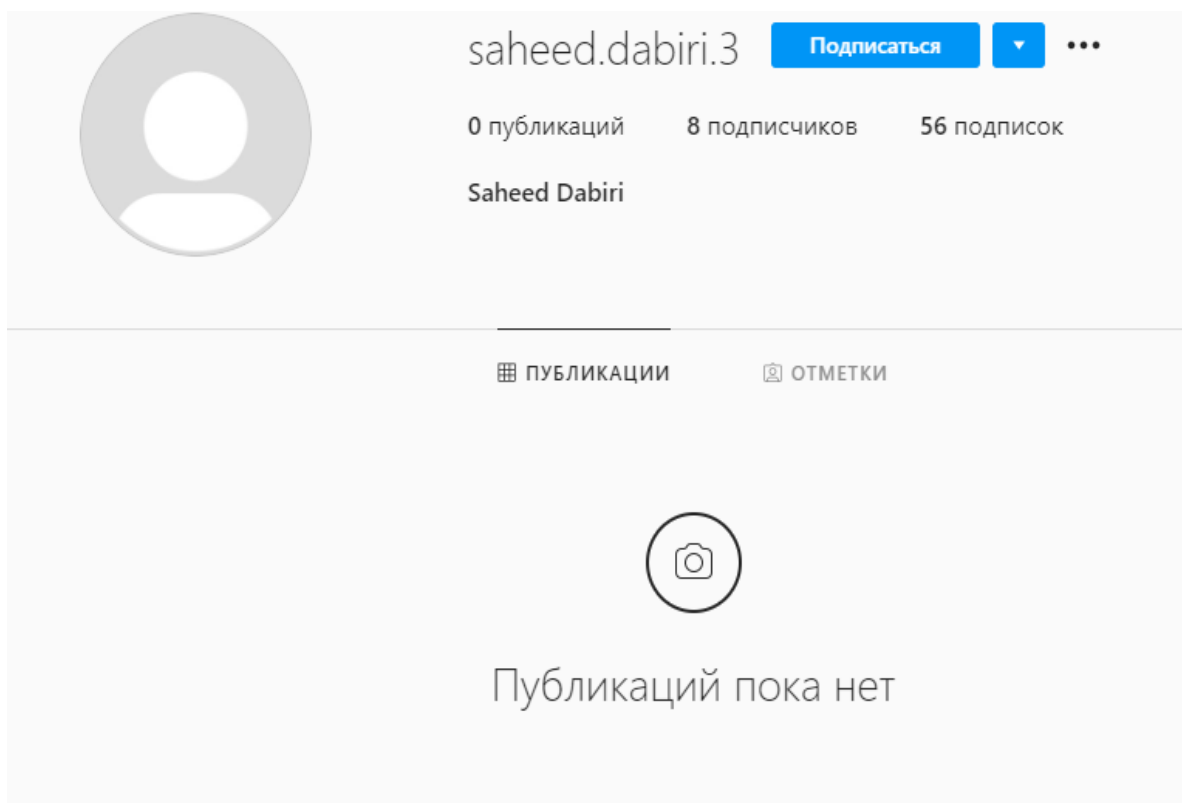


Рисунок 1.14 – Профіль у якого відсутнє фото користувача

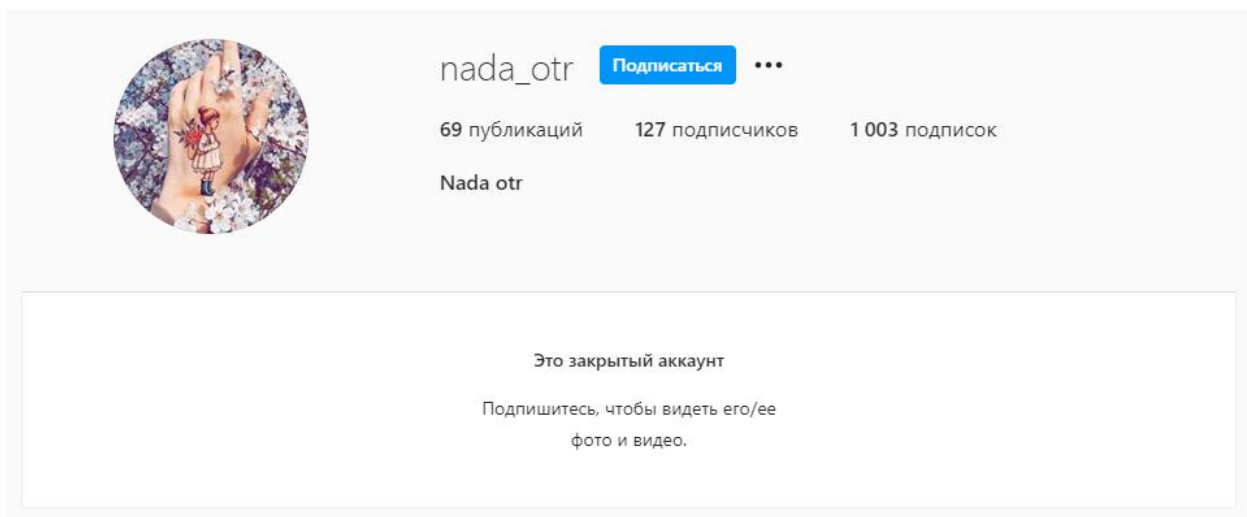


Рисунок 1.15 – Закритий аккаунт



Рисунок 1.16 – Замало постів у користувача

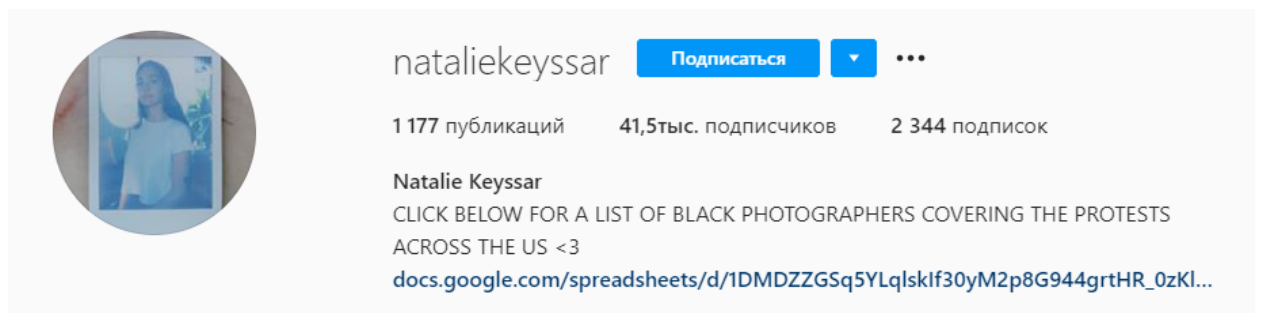


Рисунок 1.17 – Забагато підписок у користувача

3.2 Масфоловінг та маслайкінг

Через обмеження Instagram, для аккаунтів певного віку задаються різні значення для:

Кількість лайків на годину.

Кількість лайків на добу.

Кількість підписок на годину.

Кількість підписок на добу.

Час між лайками.

Час між підписками.

Програмний продукт випадково обирає 2 пости(рис. 1.18), для лайків(від 1 до 6). Після кожного лайку програма чекає певний час(Час між лайками), потім коли лайкі проставлені(рис. 1.19), відбувається підписка на користувача(рис. 1.20).

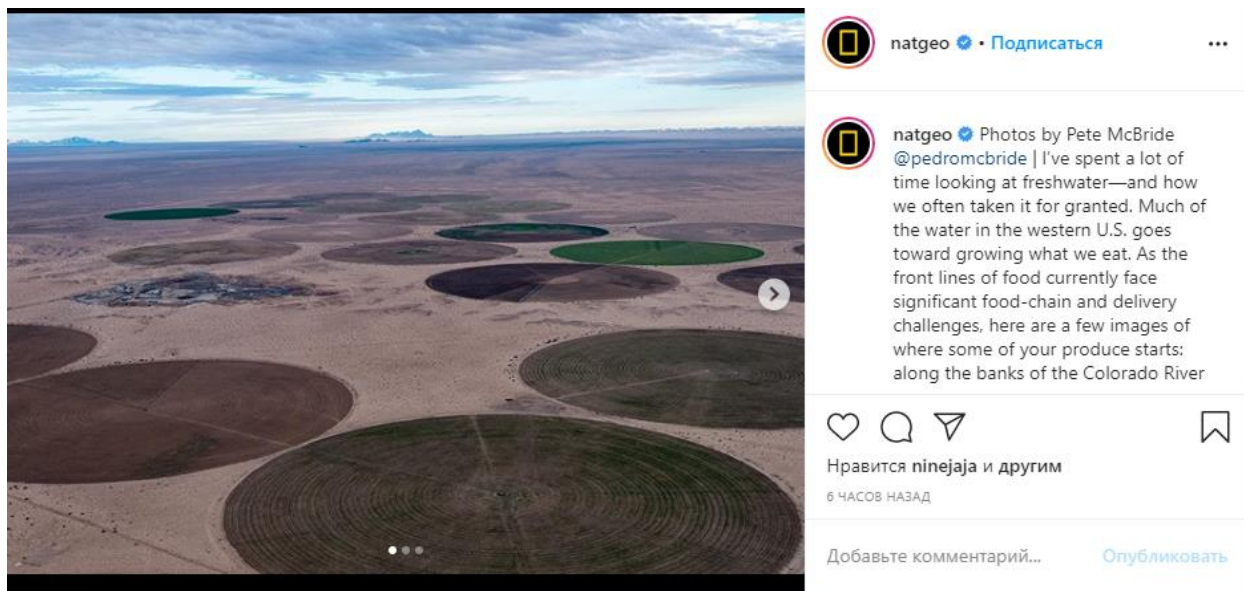


Рисунок 1.18 – Випадковий пост користувача

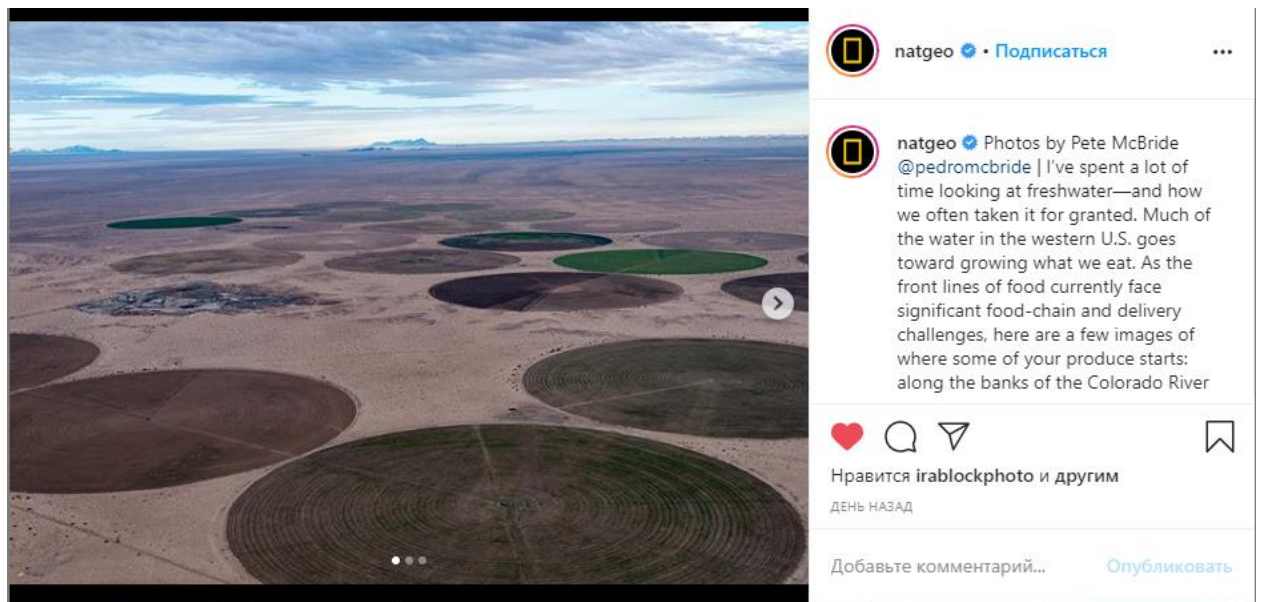


Рисунок 1.19 – Лайк на випадковий пост

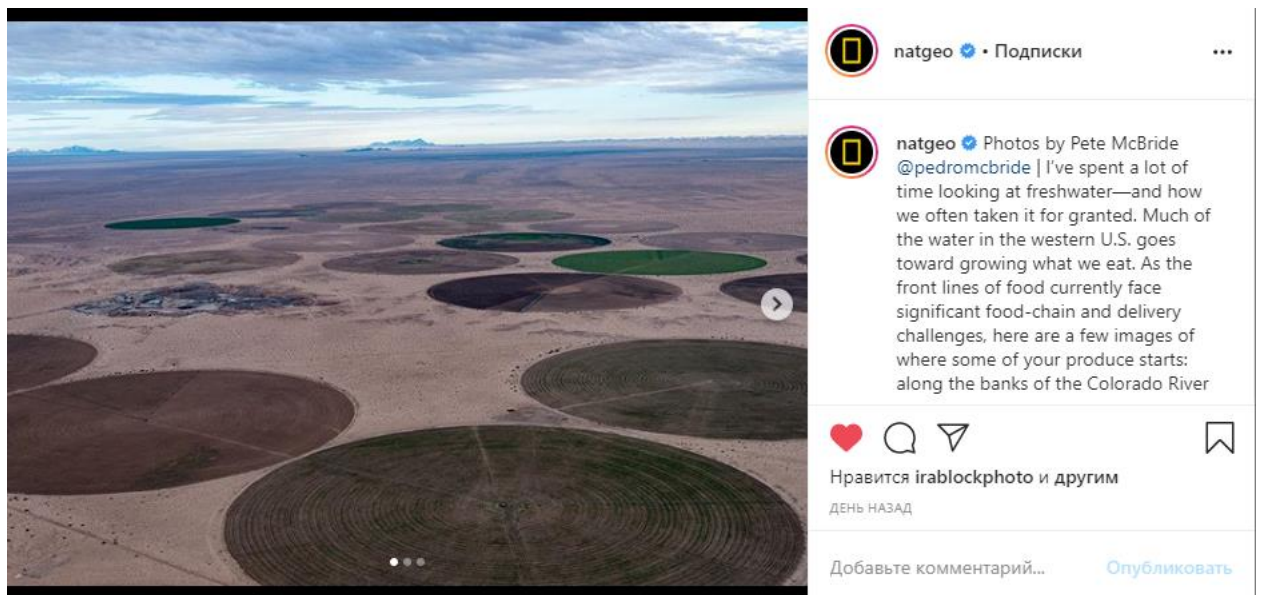


Рисунок 1.20 – Підписка на користувача

3.3 Використані методи

Щоб забезпечити максимально ефективну роботу програмного продукту, кожен потрібний елемент на сторінці перевіряється на наявність за допомогою XPath(рис 1.21).

Використовуючи XPath додаток збирає всі необхідні дані про користувача, знаходить необхідні елементи на сторінці.

3.3.1 Пошук по XPath

XPath - це мова, що використовується для пошуку вузлів дерева XML-документа. Оскільки в основі HTML може лежати структура XML (XHTML), користувачам Selenium надається можливість за допомогою цієї потужної мови відшукувати елементи в їх веб-додатках. XPath виходить за рамки простих методів пошуку за атрибутами id або name (і в той же час підтримує їх), і відкриває спектр нових можливостей, таких як пошук третього чекбокса (checkbox) на сторінці, наприклад.

Одна з вагомих підстав використовувати XPath є в наявності ситуацій, коли ви не можете володіти придатними в якості показчиків атрибутами, такими як id або name, для елемента, який ви хочете знайти. Ви можете використовувати XPath для пошуку елемента як по абсолютному шляху (не рекомендується), так і за відносним (для елементів із заданими id або name). XPath показники в тому числі можуть бути використані для визначення елементів за допомогою атрибутів відмінних від id і name.

Абсолютний шлях XPath містить в собі всі вузли дерева від кореня (html) до необхідного елемента, і, як наслідок, схильний до помилок в результаті найменших коригувань вихідного коду сторінки. Якщо знайти найближчий елемент з атрибутами id або name (в ідеалі один з елементів-батьків), можна визначити шуканий елемент, використовуючи зв'язок «батько-підлеглий». Ці зв'язки будуть куди більш стабільним і зроблять тести стійкими до змін у вихідному коді сторінки.

Елемент X може бути визначений наступним чином:

- Абсолютний шлях (поламається при щонайменшій зміні структури HTML сторінки)
- Перший елемент X в сторінці HTML
- Елемент X, для якого визначено атрибут з ім'ям id і значенням loginForm

Елемент X може бути знайдений так:

- Перший елемент X з дочірнім елементом input, для якого визначено атрибут з ім'ям name і значенням username
- Перший дочірній елемент input елемента X, для якого визначено атрибут з ім'ям id і значенням loginForm
- Перший елемент input, для якого визначено атрибут з ім'ям name і значенням username

```
def xpath_existence(url):  
    try:  
        browser.find_element_by_xpath(url)  
        existence = 1  
    except NoSuchElementException:  
        existence = 0  
    return existence
```

Рисунок 1.21 – Використання XPath

Основним методами пошуку по XPath є звернення до вузлів(nodes) або вираховувати значення XML та HTML. CSS селектори користуються подібними функціями, але XPath головною перевагою XPath слугує те, що він дозволяє робити набагато більше.

Використовуючи XPath можна збирати дані на принципі в який входить основа текстового документу, при цьому не тільки в структурі сторінки. Навіть, якщо потрібно зібрати дані з сайту з поганою структурою, XPath допомагає зробити це набагато швидше.

XML/HTML документи представляються за допомогою мови XPath у вигляді дерева елементів(вузлів). Корінний вузол(Корінний вузол представляє з себе батьківський вузол початкового елементу документа, але root) не використовується, як частина елементу.

Приклад XPath дерева:

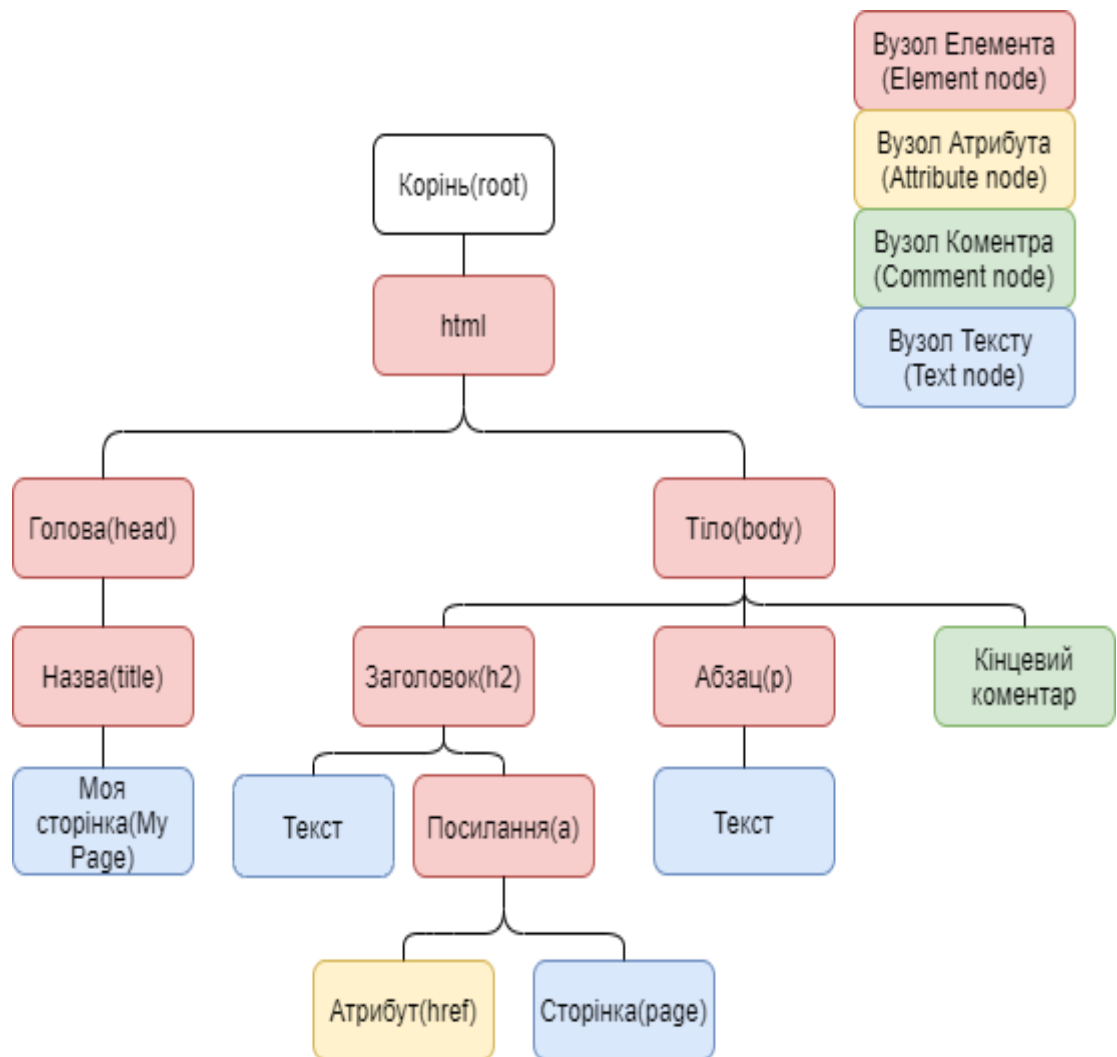


Рисунок 1.22 XPath дерево

Дерево XPath містить декілька типів вузлів:

- Вузол елемента це HTML-елемент, приклад, Посилання(тег a).
- Вузол атрибута це атрибут елемента, приклад атрибут href в тегу Атрибут/
- Вузол коментарів це коментарі в елементі, приклад, Кінцевий коментар.
- Текстовий вузол це текст всередині елементу, приклад Сторінка(page).

Питання розуміння різниці між вузлами є дуже важливим.

Далі представлено, як за допомогою XPath виділити елемент:

```
html/body/p
```

Представлені вище шляхи мають назву location path(шлях локації). Це робить можливим вказання шляху відносно контекстного вузла (прикладом на рисунку вище є корінний вузол). При слідуванні зазначеним шляхом є 3 етапи. Починається слідування з елемента html, далі в середині першого елемента знаходиться тіло(body) і вже всередині можна знайти елемент абзац(p). На кожному етапі контекстний вузол змінюється.

3.3.2 Використання XML

Мова XML завжди слугує описом документа . Структуру дерева часто називають XML-деревом і це відіграє важливу роль для легкого опису будь-якого документа XML.

Структура дерева містить кореневі (батьківські) елементи, дочірні елементи тощо. Використовуючи структуру дерева, можна ознайомитись з усіма наступними гілками та підгілками, починаючи з кореня. Розбір починається в корені, потім рухається вниз по першій гілці до елемента, береться звідти першою підгілкою і так далі до текстових вузлів.

Наступний приклад демонструє просту структуру дерева XML

```
<?xml version = "1.0"?>
<Company>
  <Employee>
    <FirstName>Henry</FirstName>
    <LastName>Ford</LastName>
    <ContactNo>111222333</ContactNo>
    <Email>ford@abc.com</Email>
    <Address>
      <City>Kiev</City>
      <State>Oblast</State>
      <Zip>110258</Zip>
    </Address>
  </Employee>
</Company>
```

Наступна структура дерева представляє вищевказаний документ XML

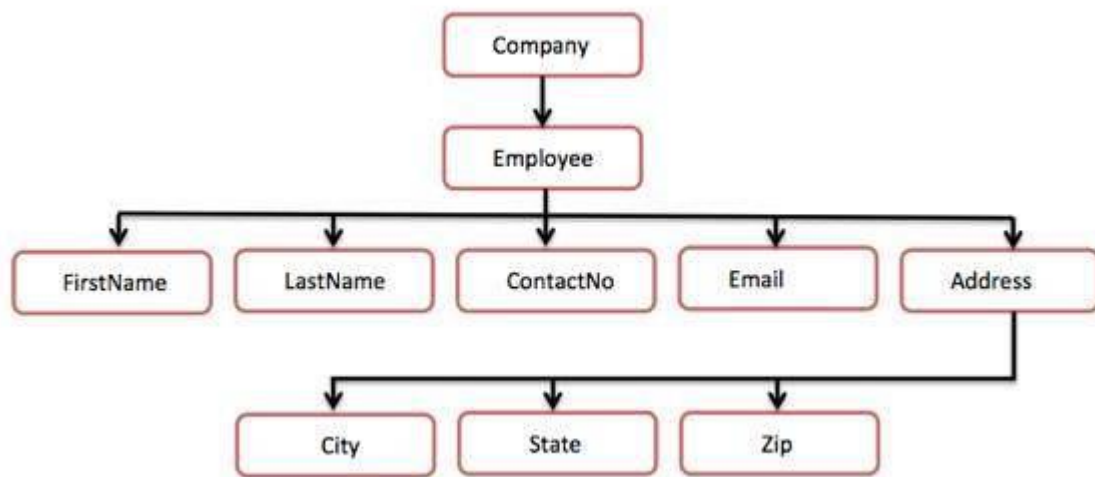


Рисунок 1.23 Приклад діаграми XML дерева

На наведеній діаграмі є кореневий елемент, названий як <компанія>. Всередині цього є ще один елемент <Робітник>. Всередині елемента службовця є п'ять гілок з назвою <Ім'я>, <Прізвище>, <Контаки>, <Почта> та <Адреса>. Всередині елемента <Адреса> є три підгалузі, названі <Місто> <Область> і <Почтовий індекс>.

3.3.3 Інші методи

Для автоматизованої авторизації(рис. 1.22) аккаунту Instagram використовуються наступні методи:

`get()`

Метод відкриває потрібну нам сторінку в браузері, приймаючи параметром URL-адресу.

`send_keys()`

Метод вводить послідовність ключів у елементі DOM, навіть якщо зустрічається послідовність ключа модифікатора. Ось список можливих натискань клавіш, які підтримує WebDriver.

Selenium click button

`click()`

Selenium може автоматично натискати на кнопки, які з'являються на веб-сторінці. Цей метод дозволяє також підписуватись та ставити лайки користувачам автоматично.

```
find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[2]/div/label/input").send_keys(login)
find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[3]/div/label/input").send_keys(password)
find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[4]").click()
```

Рисунок 1.24 – Автоматизована авторизація

Методом `get()` відкривається сторінка потрібного користувача і за допомогою XPath ми отримуємо доступ до бази його підписників(рис. 1.23).

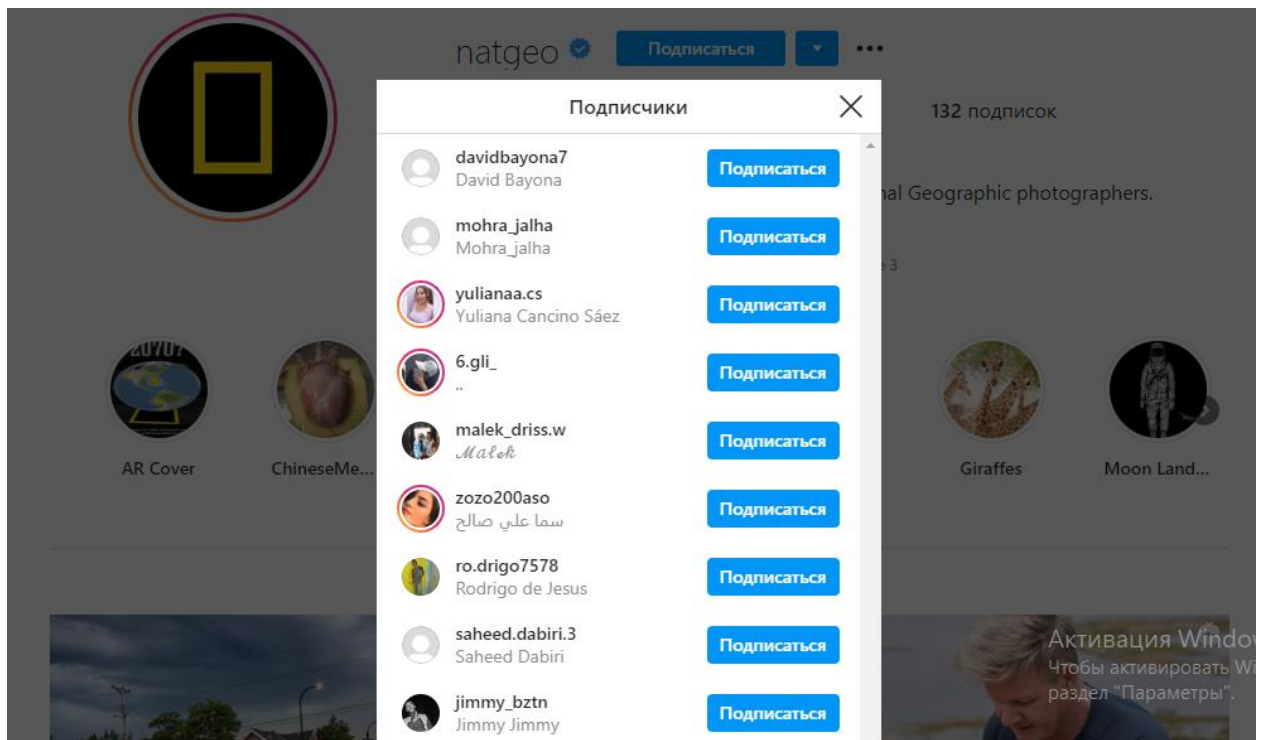


Рисунок 1.25 – Підписники аккаунта

За допомогою метода `execute_script()` ми прокручуємо підписників у відкритому вікні та зберігаємо до файлу за допомогою `open()`. Також за допомогою методу `open()` програма зберігає відфільтрований список користувачів, а також після підписки видаляє їх звідти і вже заносить до текстового файлу з підписками.

Time - доступ до часу та перетворення

Цей модуль надає різні функції, пов'язані з часом. Також модулі дати та календаря.

Функції модулю `time` необхідні для затримок між діями парсеру, що в свою чергу забезпечує плавну роботу і оберігає бота від підозр Instagram в автоматизованому користуванні соціальною мережею.

re - Регулярні операції вираження

Цей модуль забезпечує регулярні операції зіставлення виразів.

Синтаксис регулярних виразів

Регулярний вираз (або RE) задає набір рядків, який відповідає йому; функції в цьому модулі дозволяють вам перевірити, чи відповідає певна рядок заданому регулярному виразу (або чи відповідає певний регулярний вираз певній рядку, що зводиться до тієї ж речі).

Методи модуля `re` допомагають видалити непотрібні написи з шуканих елементів.

Іменні функції, інструкція `def`

Функція в `python` - об'єкт, який приймає аргументи і повертає значення. Зазвичай функція визначається за допомогою інструкції `def`.

`random` - Створення псевдовипадкових чисел

Цей модуль реалізує генератори псевдовипадкових чисел для різних розподілів.

Для цілих чисел існує рівномірний вибір з діапазону. Для послідовностей є рівномірний вибір випадкового елемента, функція генерування випадкової перестановки на місці списку та функція для випадкової вибірки без заміни.

За допомогою модулю `random` вибираються випадкові пости для лайкінгу.

Використано винятки:

- `NoSuchElementException`;
- `StaleElementReferenceException`;
- `ElementClickInterceptedException`.

`NoSuchElementException`

Цей виняток піднімається, коли елемент не вдалося знайти.

`StaleElementReferenceException`

Цей виняток піднімається, коли посилання на елемент тепер "застаріло" - означає, що елемент був доступний під час операції пошуку, але більше не з'являється у DOM сторінки під час виконання певної операції.

ElementClickInterceptedException

Цей виняток піднімається, коли елемент не є клікабельним.

```
from selenium.common.exceptions import NoSuchElementException  
from selenium.common.exceptions import StaleElementReferenceException  
from selenium.common.exceptions import ElementClickInterceptedException
```

Рисунок 1.26 – Використання винятків в програмному коді

3.4 Керівництво користувача

При роботі першого модулю `create_persons_list.py` термінал будуть виведені повідомлення, в яких буде відображено, що користувач за допомогою клавіатури повинен ввести такі значення:

- Логін;
- Пароль;
- Посилання на сторінку для збору даних;
- Мінімальна кількість користувачів.

Важливим є щоб користувач сам визначив сторінку для збору даних, адже це призведе до найкращої якості отриманої ЦА.

При роботі другого модулю `newfiltr.py` термінал будуть виведені повідомлення, в яких буде відображено, що користувач за допомогою клавіатури повинен ввести такі значення:

- Максимальна кількість днів від останньої публікації;
- Максимальна кількість підписок у аккаунта;
- Мінімальна кількість публікацій у аккаунта.

При роботі третього модулю liker.ru термінал будуть виведені повідомлення, в яких буде відображено, що користувач за допомогою клавіатури повинен ввести такі значення:

- Час між лайками;
- Лайків за годину;
- Лайків за добу;
- Підписок за добу;
- Підписок за годину.

При виборі цих значень слід обов'язково скористатися інформацією про дослідження можливої кількості лайків та підписок в залежності від часу створення аккаунта, яку дослідив та привів вище.

3.5 Можливі вдосконалення

Існує ряд труднощів, які можуть досягнути замовника подібного роду послуг при роботі з додатком. Найпоширеніші з них:

- Можливість блокування облікового запису. Користувач ризикує бути заблокований через підозрілу активність. Справа в тому, що захисний алгоритм соціальної мережі спрямований на боротьбу з автоматизованими способами просування профілю. Вирішити дану проблему досить просто - використовуйте спеціально створену для активного автоматизованого збору даних сторінку.

- Боти і профілі комерційної спрямованості. При зборі бази цільової аудиторії враховуються всі задані критерії, але іноді під цей опис потрапляють аккаунти конкурентів та боти. Для вдосконалення, можливо ще додавання деяких критеріїв для збору аудиторії.

ВИСНОВКИ

Результати:

- досліджено активний режим збору, при якому ми самостійно ініціюємо запит на інформацію, в якій ми зацікавлені.
- досліджено сучасний метод активного автоматизованого збору даних за допомогою пошуку елементів на сторінці по XPath.
- розроблено програмне забезпечення(бот), яке активно автоматизовано збирає дані про аккаунти в Instagram, відфільтровує небажані аккаунти(Користувач задає потрібні параметри) та забезпечує приток нових підписників, котрі можуть бути зацікавлені у тому, що пропонує користувач бота.

ПЕРЕЛІК ПОСИЛАНЬ

1. <https://www.publichealth.columbia.edu/research/population-health-methods/content-analysis>
2. <https://www.scribbr.com/methodology/content-analysis/>
3. <https://lpgenerator.ru/blog/2019/02/28/instagram-v-cifrah-statistika-na-2020-god/>
4. <https://www.quora.com/What-is-a-personal-blog-on-Instagram#:~:text=Instagram%20is%20a%20picture%20and,a%20form%20of%20personal%20blogging.>
5. <https://instazood.com/blog/the-history-of-instagram/>
6. <https://instasec.ru/limity-i-ogranicheniya-v-instagram/>
7. <https://instasec.ru/skolko-laykov-v-den-mozhno-stavit-v-instagram/>
8. <https://instasec.ru/skolko-v-den-mozhno-podpisyvatsya-v-instagram/>
9. https://en.wikipedia.org/wiki/Artificial_intelligence
10. <https://habr.com/ru/post/348314/>
11. https://www.sas.com/en_us/insights/analytics/what-is-artificial-intelligence.html
12. <https://www.britannica.com/technology/artificial-intelligence>
13. <https://instaplust.me/blog/parsing/>
14. <https://zen-promo.com/blog/post/detailed-overview-of-instagram-parsing-service-how-to-parse-the-target-audience>
15. <https://www.promptcloud.com/scrape-data-from-instagram/>
16. <https://blog.apify.com/scrape-instagram-posts-comments-and-more-21d05506aeb3>
17. <https://builtin.com/artificial-intelligence>
18. <https://futureoflife.org/background/benefits-risks-of-artificial-intelligence/>
19. <https://searchenterpriseai.techtarget.com/definition/AI-Artificial-Intelligence>
20. <https://www.pythonforbeginners.com/learn-python/what-is-python/>
21. <https://opensource.com/resources/python>
22. <https://selenium-python.com/>
23. <https://docs.python.org/2/library/webbrowser.html>
24. <https://habr.com/ru/post/250975/>
25. <https://pythonspot.com/selenium-click-button/>
26. <https://docs.python.org/3/library/time.html>
27. <https://docs.python.org/3/library/re.html>
28. <https://selenium-python.readthedocs.io/ waits.html>
29. <https://docs.python.org/3/library/random.html>
30. <https://pythonworld.ru/typy-dannyx-v-python/fajly-rabota-s-fajlami.html>

31. <http://allselenium.info/top-5-commonly-encountered-exceptions-in-python-selenium-webdriver/>
32. <https://www.templatemonster.com/blog/got-banned-by-instagram-what-to-do/>

Додаток 1. Специфікація

Позначення	Найменування	Примітки
Документація		
1	Пояснювальна записка	н/п
2	Додаток 2. Текст програмного модулю	н/п
3	Додаток 3. Опис програмного модулю	н/п
Компоненти		
1	Пошук елемента (XPath)	XPath - це мова запитів до елементів xml або xhtml документа. XPath є декларативною мовою запитів. Щоб отримати дані, що цікавлять, необхідно всього лише створити запит, що описує ці дані.
2	Метод відкриття випадкових постів	Метод надає можливість для генерації випадкового вибору елементів послідовності. Обирається кілька випадкових постів на сторінці користувача.
3	Метод очищення даних	Задача методу полягає у виявленні та видаленні не потрібних даних з основною метою поліпшення якості даних.

4	Метод очікування	Неявне очікування методу повідомляє WebDriver для запитування DOM протягом певного часу, намагаючись знайти будь-який елемент (або елементи), недоступні одразу.
7	Метод відображення текстових даних	Метод полягає у візуальному представленні списків відфільтрованих користувачів.
8	Метод доступу до часу	Метод полягає у тому, щоб призупинити виконання нитки виклику на задану кількість секунд. Фактичний час призупинення може бути меншим, ніж вимагається.

Додаток 2. Лістинг коду додатку

create_persons_list.py

```
from selenium import webdriver
import time

# параметри
login = input("Логін: ")
password = input("Пароль: ")
url = input("Посилання на сторінку для збору даних: ")
all = input("Мінімальна кількість користувачів: ")

browser = webdriver.Chrome("E:\Instagram bot\chromedriver.exe")
browser.implicitly_wait(5)

# Вхід в аккаунт
browser.get("https://www.instagram.com/accounts/login/")
browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[2]/div/label/input").send_keys(login)
browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[3]/div/label/input").send_keys(password)
browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[4]").click()
time.sleep(5)

# Відкривання заданої сторінки та її підписників
browser.get(url)
time.sleep(5)
browser.find_element_by_xpath("//section/main/div/header/section/ul/li[2]/a").click()
time.sleep(5)
element = browser.find_element_by_xpath("/html/body/div[4]/div/div[2]")

# Плавне прокручування підписників
browser.execute_script("arguments[0].scrollTop = arguments[0].scrollHeight/%s" % 6, element)
time.sleep(0.8)
browser.execute_script("arguments[0].scrollTop = arguments[0].scrollHeight/%s" % 4, element)
time.sleep(0.8)
browser.execute_script("arguments[0].scrollTop = arguments[0].scrollHeight/%s" % 3, element)
time.sleep(0.8)
```

```

browser.execute_script("arguments[0].scrollTop = arguments[0].scrollHeight/%s"
%2, element)
time.sleep(0.8)
browser.execute_script("arguments[0].scrollTop = arguments[0].scrollHeight/%s"
%1.4, element)
time.sleep(0.8)

# Створення динамічного масиву для запису підписників користувача
pers = []
t = 0.7
num_scroll = 0
p = 0

# Запис в масив
while len(pers) < all :
    num_scroll += 1
    browser.execute_script("arguments[0].scrollTop = arguments[0].scrollHeight", e
lement)

    if num_scroll % 10 == 0 :
        print("!")
        persons = browser.find_elements_by_xpath("//div[@role='dialog']/div[2]/ul/d
iv/li/div/div/div/div/a[@title]")
        for i in range(len(persons)):
            pers.append(str(persons[i].get_attribute("href")))
        time.sleep(t)

    if(len(pers) > (2000 + 1000*p) ):
        print("\Очікуйте 10 хвилин")
        time.sleep(60*10)
        p += 1

# Запис до файлу
f = open("E:\Instagram bot\pers_list.txt", "w")
for person in pers:
    f.write(person)
    f.write("\n")
f.close()

# Закриття браузера
browser.quit()

```

newfiltr.py

```
from selenium import webdriver
from datetime import timedelta, datetime
import time
import re

# Параметри
days = input("Максимальна кількість днів від останньої публікації: ") # максимальна кількість днів від останньої публікації
subs = input("Максимальна кількість підписок у аккаунта: ") # максимальна кількість підписок у аккаунта
publ = input("Мінімальна кількість публікацій у аккаунта: ") # мінімальна кількість публікацій у аккаунта

today = datetime.now()

from selenium.common.exceptions import NoSuchElementException
from selenium.common.exceptions import StaleElementReferenceException

# Функція перевіряє чи є елемент на сторінці
def xpath_existence(url):
    try:
        browser.find_element_by_xpath(url)
        existence = 1
    except NoSuchElementException:
        existence = 0
    return existence

browser = webdriver.Chrome("E:\Instagram bot\chromedriver.exe")
#browser.implicitly_wait(5)

#browser.get("https://www.instagram.com/accounts/login/")
#browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[2]/div/label/input").send_keys(login)
#browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[3]/div/label/input").send_keys(password)
#browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[4]").click()
#time.sleep(5)

# Відкрити та записати файл в масив
f = open("E:\Instagram bot\pers_list.txt", 'r')
```

```

file_list = []
for line in f:
    file_list.append(line)
f.close()
print(file_list)

# Створення динамічного масиву відфільтрованого списку
filtered_list = []
i = 0
j = 0

# Відкриття сторінок з файлу
for person in file_list:
    j += 1
    #browser.get("https://www.instagram.com/accounts/login/")
    #browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form
/div[2]/div/label/input").send_keys(login)
    #browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form
/div[3]/div/label/input").send_keys(password)
    #browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form
/div[4]").click()
    #time.sleep(3)
    browser.get(person)
    time.sleep(1.5)

    # Перевірка на приватність
    element = "//section/main/div/div/article/div[1]/div/h2"
    if xpath_existence(element) == 1:
        try:
            if browser.find_element_by_xpath(element).text == "This Account is Private" or "Это закрытый аккаунт":
                print(j, "Приватний аккаунт")
                continue
        except StaleElementReferenceException:
            print("Помилка 1")

    # Перевірка підписок
    element = "//section/main/div/header/section/ul/li[3]/a/span"
    if xpath_existence(element) == 0:
        print(j, "Помилка 2")
        continue
    status = browser.find_element_by_xpath(element).text
    status = re.sub(r'\s', "", status)
    if int(status) > subs:

```



```

print(j, "Забагато підписок")
continue

# Перевірка чи є сторонні посилання
element = "//section/main/div/header/section/div[2]/a"
if xpath_existence(element) == 1:
    print(j, "Є сторонні посилання")
    continue

# Перевірка кількості публікацій
element = "//section/main/div/header/section/ul/li[1]/a/span"
if xpath_existence(element) == 0:
    print(j, "Помилка 3")
    continue
status = browser.find_element_by_xpath(element).text
status = re.sub(r'\s', "", status)
if int(status) < publ:
    print(j, "Замало публікацій")
    continue

# Перевірка фото профілю
element = "//section/main/div/header/div/div/span/img"
if xpath_existence(element) == 0:
    print(j, "Помилка 4")
    continue
status = browser.find_element_by_xpath(element).get_attribute("src")
#print(status)
#if status.find("44884218_345707102882519_2446069589734326272_n.jpg") =
= 1:
if status.find("s150x150") == -1:
    print(j, "Відсутнє фото профілю")
    continue

# Перевірка дати останньої публікації
element = "//a[contains(@href, '/p/')] "
if xpath_existence(element) == 0:
    print(j, "Помилка")
    continue
status = browser.find_element_by_xpath(element).get_attribute("href")
browser.get(status)
post_date = browser.find_element_by_xpath("//time").get_attribute("datetime")
year = int(post_date[0:4])
month = int(post_date[5:7])
day = int(post_date[8:10])

```

```
post_date = datetime(year, month, day)
period = today - post_date
if period.days > days:
    print(j, "Остання публікація занадто давно")
    continue

# Запис в відфільтрований список
filtered_list.append(person)
print(j, "Аккаунт доданий до списку")
i += 1

if i > 2:
    break

# Вихід із циклу

# Запис до файлу відфільтрованого списку
f = open("E:\Instagram bot\pers_new_list.txt", "w")
for line in filtered_list:
    f.write(line)
f.close()

# Закриття браузера
browser.quit()
```

liker.py

```
from selenium import webdriver
import time
import random
random.seed()

# Параметри
like_time = input("час між лайками: ") # - час між лайками
all_likes = input("лайків за добу: ") # - лайків за добу
all_subs = input("підписок за добу: ") # - підписок за добу
hour_like = input("лайків за годину: ") # - лайків за годину
hour_sub = input("підписок за годину: ") # - підписок за годину

likes = 0 # - скільки лайків вже проставлено
subs = 0 # - скільки вже підписок зроблено

# Винятки
from selenium.common.exceptions import NoSuchElementException
from selenium.common.exceptions import StaleElementReferenceException
from selenium.common.exceptions import ElementClickInterceptedException

# Функція перевіряє чи є елемент на сторінці
def xpath_existence(url):
    try:
        browser.find_element_by_xpath(url)
        existence = 1
    except NoSuchElementException:
        existence = 0
    return existence

browser = webdriver.Chrome("E:\Instagram bot\chromedriver.exe")
browser.implicitly_wait(5)

# Вхід в аккаунт
browser.get("https://www.instagram.com/accounts/login/")
time.sleep(2)
browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div[2]/div/label/input").send_keys(login)
```

```

browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div
[3]/div/label/input").send_keys(password)
browser.find_element_by_xpath("//section/main/div/article/div/div[1]/div/form/div
[4]").click()
time.sleep(5)

# Запис відфільтрованого списку в динамічний масив
file_list = []
f = open("E:\Instagram bot\pers_new_list.txt", "r")
for line in f:
    file_list.append(line)
f.close()
print(file_list)

# Створення списку моїх підписок
subs_list = []
f1 = open("E:\Instagram bot\my_subs.txt", "r")
for line in f1:
    subs_list.append(line)
f1.close()
print(subs_list)

j = 0
n = 0
next_person = 0
start_time = time.time()
print(start_time)

# Цикл
for person in file_list:

    # Умови виходу із циклу
    if likes >= all_likes:
        print("Максимальна кількість лайків за добу")
        break
    if subs >= all_subs:
        print("Максимальна підписок лайків за добу")
        break

    # Максимальна кількості підписок за годину
    if((time.time() - start_time) <= 60*60) and (hour_sub <= subs):
        print("Максимальна підписок за годину")
        print("Зачекайте", int((60*60 - (time.time() - start_time))/60), "хв")

```

```

# Видалення з відфільтрованого листа того,на кого вже підписався
f = open("E:\Instagram bot\pers_new_list.txt", "w")
for i in range(j, len(file_list)):
    f.write(file_list[i])
f.close()

time.sleep(60*60 - (time.time() - start_time))
start_time = time.time()
subs = 0
likes = 0

# Максимальна кількість лайків за годину
if((time.time() - start_time) <= 60*60) and (hour_like <= likes):
    print("Максимальна лайків за годину")
    print("Зачекайте", int((60*60 - (time.time() - start_time))/60), "хв")

# Видалення з відфільтрованого листа того,на кого вже підписався
f = open("E:\Instagram bot\pers_new_list.txt", "w")
for i in range(j, len(file_list)):
    f.write(file_list[i])
f.close()

time.sleep(60*60 - (time.time() - start_time))
start_time = time.time()
subs = 0
likes = 0

# Обнуління часу
if( (time.time() - start_time) >= 60*60):
    start_time = time.time()
    subs = 0
    likes = 0

# Порівняння з масивом підписок
for line in subs_list:
    next_person = 0
    if person == line:
        next_person = 1
        print(j + 1, "\tВи вже підписані на цей аккаунт")
        j += 1
        n += 1
        break
    if next_person == 1:
        continue

```

```

# Вивод номеру в термінал
j += 1
print("\n" + str(j - n) + ": ")

# Відкриття сторінок зі списку
browser.get(person)
time.sleep(1.5)

# Відкриття публікацій та лайки

# Перевірка чи є підписка
element = "//section/main/div/header/section/div[1]/div[1]/span/span[1]/button"
if xpath_existence(element) == 1:
    try:
        follow_status = browser.find_element_by_xpath(element).text
    except StaleElementReferenceException:
        print(j, "Помилка 1")
        continue
    if (follow_status == "Following") or (follow_status == "Подписки"):
        print(j, "Вже підписані на цей аккаунт\n")
        continue

# Пошук 2 публікацій та лайк
element = "//a[contains(@href, '/p/')] "
if xpath_existence(element) == 0:
    print(j, "Помилка 2")
    continue
posts = browser.find_elements_by_xpath(element)
i = 0
for post in posts:
    posts[i] = post.get_attribute("href")
    i += 1
rand_post = random.randint(0,5)
for i in range(2):
    browser.get(posts[rand_post + i])
    time.sleep(0.5)
    browser.find_element_by_xpath("//section/main/div/div/article/div[2]/section[1]/span[1]/button").click()
    likes += 1
    print("+1 лайк")
    time.sleep(like_time)

# Підписка

```

```

try:
    element = "//section/main//div/div/article/header/div[2]/div[1]/div[2]/button
"

    if xpath_existence(element) == 0:
        print(j, "Помилка 3.1")
    try:
        browser.find_element_by_xpath(element).click()
    except StaleElementReferenceException:
        print(j, "Помилка 3.2")
        continue
    except ElementClickInterceptedException:
        print(j, "Помилка 3.3")
        continue

    subs += 1
    print("+1 підписка", person[0:len(person) - 1])
    time.sleep(0.5)

# Запис до файлу підписок
f = open("E:\Instagram bot\my_subs.txt", "a")
f.write(person)
f.close()

# Вихід із цикла

# Видалення зі списку відфільтрованих аккаунтів, на кого підписався
f = open("E:\Instagram bot\pers_new_list.txt", "w")
for i in range(j, len(file_list)):
    f.write(file_list[i])
f.close()

# Вихід із браузера
browser.close()

```

Додаток 3. Опис програмного модуля

АНОТАЦІЯ

Баун Б. «Активний автоматизований збір даних(на прикладі бота, який збирає дані про користувачів соціальної мережі Instagram з подальшим залученням деяких до активності власного профілю)». КПІ ім. Ігоря Сікорського, Київ, 2020.

В ході проведеної роботи було розроблено додаток, що збирає дані про підписників деякої сторінки в Instagram, фільтрує цих користувачів за певними критеріями і залучає до активності на профілі користувача, який використовує даний додаток.

Реалізована система вирішує проблему аналізу даних користувачів з використанням відповідного методу пошуку та дослідження елементів на сторінці.

ЗМІСТ

1 ЗАГАЛЬНІ ВІДОМОСТІ ТА ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ	4
2 ОПИС ЛОГІЧНОЇ СТРУКТУРИ	5
3 ВИКОРИСТАНІ ТЕХНІЧНІ ЗАСОБИ.....	8
4 ВХІДНІ ТА ВИХІДНІ ДАНІ	8

1. ЗАГАЛЬНІ ВІДОМОСТІ ТА ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ

Python - мова програмування загального призначення, що є ще одним способом сказати, що його можна використовувати майже для всього.

Найголовніше, що це інтерпретована мова, що означає, що написаний код насправді не переводиться на машиночитаний формат під час виконання.

Тоді як більшість мов програмування здійснюють цю конверсію ще до того, як програма навіть буде запущена. Цей тип мови також називають «мовою сценаріїв», оскільки він спочатку мав використовуватись для тривіальних проектів.

Мова програмування Python останнім часом все частіше використовується для аналізу даних, як в науці, так і комерційній сфері. Цьому сприяє простота мови, а також велика різноманітність відкритих бібліотек.

Останніми роками ряд бібліотек досяг рівня, що дозволяє користувачам R скористатися простотою, гнучкістю та продуктивністю Python, не приносячи шкоди функціоналу, який ці старі програми накопичували протягом багатьох років.

Причини такої популярності мови Python:

- Об'єктивно-орієнтована мова;
- Велика кількість розширених;
- Простота в розумінні і дослідженнях;
- Завдяки таким пакетам, як pandas, numpy та scikit-learn, Python ідеально підходить для розробки алгоритмів аналізу даних.

Мова R може:

- Зберігають пакети практично для будь-якого статистичного додатку;
- На даний момент у CRAN є більше ніж 10 тис. пакетів;

- Оснащені чудово продуманими бібліотеками для візуалізації, такими як ggplot2;
- Є можливість автономного аналізу.

Вже той час мова програмування R відстає у продуктивності від Python, а

також викликає засорення пам'яті при роботі з великими наборами даних.

Оскільки додаток працює з даними про користувачів соціальної мережі Instagram та аналізує їх, робота з більш продуктивною мовою програмування позначиться на результатах в кращу сторону.

2. ОПИС ЛОГІЧНОЇ СТРУКТУРИ

Вимогами до додатку є:

- можливість обрати певний аккаунт для першого збору даних – бази підписників;
- можливість обрати параметри для збору даних вже з готової бази;
- можливість слідкувати за роботою автоматичного програмного забезпечення на кожному етапі;
- можливість зупинити роботу бота в потрібний момент.

Даний список показує, що основною складністю реалізації програмного забезпечення є алгоритм вирішення задачі (рисунки 1), а не взаємодія з користувачем. Тому, для даного додатку рішенням є обмежитися консольним інтерфейсом взаємодії з користувачем інтерфейсом взаємодії з користувачем, це задовольняє усім вимогам, що були наведені вище й добре підходить у разі необхідності реалізувати той чи інший алгоритм збору даних на практиці.

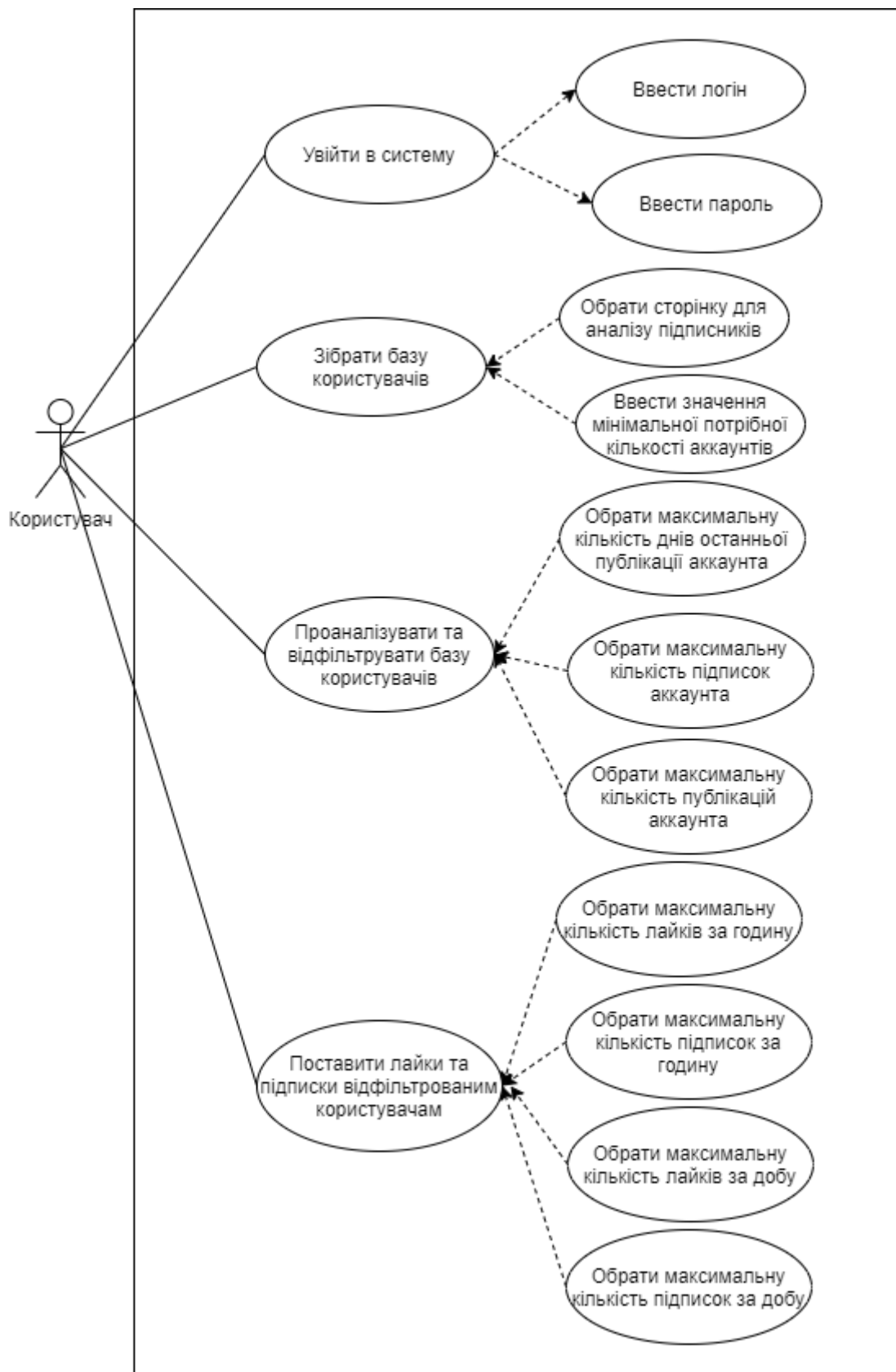


Рисунок 1 – Use Case діаграма додатку

При роботі додаток має взаємодіяти зі сторінками користувачів соціальної мережі Instagram. Для цього він зчитує такі дані зі сторінки (фото профілю, кількість підписок, підписними, дата публікацій, тип аккаунта, наявність сторонніх посилань).

```
def xpath_existence(url):  
    try:  
        browser.find_element_by_xpath(url)  
        existence = 1  
    except NoSuchElementException:  
        existence = 0  
    return existence
```

Дана функція дозволяє перевірити чи наявний потрібний елемент на сторінці, внутрішня функція `find_element_by_xpath()` отримує дані про елемент за допомогою xml або xhtml документу.

3. ВИКОРИСТОВУВАНІ ТЕХНІЧНІ ЗАСОБИ

Для відтворення роботи даного додатку було використувано ноутбук з мікропроцесором Intel(R) Core(TM) i5-6500U CPU @ 2.80 GHz, з розміром RAM 4 гігабайти та з 64-бітовою операційною системою Windows 10.

У разі необхідності підвищення швидкості роботи додатку рекомендується застосування технічних засобів з кращими характеристиками, ніж ті, які були представлені вище.

4. ВХІДНІ ТА ВИХІДНІ ДАНІ

4.1 Вхідні дані

Вхідними даними для додатку є адреса початкової сторінки для збору даних про підписників,

```
browser.get("https://www.instagram.com/natgeo/")
```

далі файл зі зібраною базою користувачів

```
pers_list.txt
1 https://www.instagram.com/_prashil_15135/
2 https://www.instagram.com/_nihad_b1vk_/
3 https://www.instagram.com/ingaedemova/
4 https://www.instagram.com/mithun_sai_m/
5 https://www.instagram.com/peroladoamaral/
6 https://www.instagram.com/edde_indonesia/
7 https://www.instagram.com/awaaa_njaay/
```

а потім для просування використовується відфільтрована база

```
pers_new_list.txt
1 https://www.instagram.com/imsaurabh_prasad/
2 https://www.instagram.com/ayouta_beauty15/
3 https://www.instagram.com/bhrh.bgbny/
```

4.2 Вихідні дані

Вихідним даними є проаналізовані дані про користувачів, якщо сторінка не задовольняє вимоги клієнта додатку, виводиться інформація по якому критерію даний аккаунт не підходить(проілюстровано нижче).

```
1 Забагато підписок
2 Приватний аккаунт
3 Забагато підписок
4 Аккаунт доданий до списку
5 Аккаунт доданий до списку
6 Приватний аккаунт
7 Приватний аккаунт
8 Аккаунт доданий до списку
9 Є сторонні посилання
10 Замало публікацій
11 Остання публікація занадто давно
12 Забагато підписок
13 Замало публікацій
14 Замало публікацій
```

Далі додаток опрацьовує список користувачів, які задовольняють вимоги клієнта та виводить інформацію, які дії були здійснені на сторінках відфільтрованих користувачів.

```
1:
+1 лайк
+1 лайк
+1 підписка https://www.instagram.com/alicjaszostakiewicz/
2      Ви вже підписані на цей аккаунт

2:
+1 лайк
+1 лайк
+1 підписка https://www.instagram.com/wala12493/
Максимальна лайків за годину
Зачекайте 58 хв
```

В кінці додаток заносить в новий список “Мої підписки” сторінки користувачів на які було здійснена підписка.

```
≡ my_subs.txt
1 https://www.instagram.com/alicjaszostakiewicz/
2 https://www.instagram.com/wala12493/
```